
Reconocimiento de una plataforma para aterrizaje de drones mediante técnicas de Visión por Computación

CHRISTIAN GÓMEZ REDONDO

MÁSTER EN INGENIERÍA INFORMÁTICA, FACULTAD DE INFORMÁTICA,
UNIVERSIDAD COMPLUTENSE DE MADRID



Trabajo Fin Máster en Ingeniería Informática

30/06/2020

Nota: 8

Director: Segundo Esteban San Román

Recognition of a drone landing platform using Computer Vision techniques

CHRISTIAN GÓMEZ REDONDO

MÁSTER EN INGENIERÍA INFORMÁTICA, FACULTAD DE INFORMÁTICA,
UNIVERSIDAD COMPLUTENSE DE MADRID



Trabajo Fin Máster en Ingeniería Informática

30/06/2020

Director: Segundo Esteban San Román

*A todos mis familiares y amigos, que siempre
me han apoyado y ayudado en la realización
de este proyecto*

Agradecimientos

En primer lugar, quisiera dar las gracias a mi primer director de Trabajo de Fin de Máster, Gonzalo Pajares Martinsanz, por permitirme trabajar en este proyecto y por estar siempre dispuesto a facilitarme cualquier ayuda o material para el desarrollo del mismo con el fin de guiarme para conseguir mis objetivos.

Y por supuesto agradecer también al director de este proyecto, Segundo Esteban San Román, por acceder a dirigirme y guiarme en el desarrollo y finalización de este proyecto.

Mencionar también a todas las personas que han evaluado mi aplicación y han dedicado una parte de su tiempo a valorarla y probarla, especialmente a mi familia y amigos.

Resumen

El procesamiento digital de imágenes por computación es una tecnología aplicada a las imágenes digitales con el fin de mejorar la calidad o para adquirir información.

En este Trabajo Fin de Máster se busca realizar un software capaz de identificar y procesar imágenes que se encuentren encuadradas en medios con diferentes características de luz, sombras o colores, con el fin de poder ser utilizado por drones a la hora de su aterrizaje.

El software es capaz de identificar una plataforma por tres factores: colores, formas geométricas y distancia. Esta funcionalidad le permitirá al dron distinguirla bien en el medio que le rodea.

Palabras clave: UAV, Dron, Plataforma de aterrizaje, Procesamiento digital de imágenes.

Abstract

Computer digital image processing is a technology applied to digital images for the purpose of improving quality or for acquiring information.

In this Master's Final Project, we seek to make software capable of identifying and processing images that are located in media with different characteristics of light, shadow or color, in order to be able to be used by drones when landing.

The software is capable of identifying a platform by three factors: colors, geometric shapes and distance. This functionality will allow the drone to distinguish it well in the environment around it.

Keywords: UAV, Drone, Landing platform, Digital image processing

Índices de contenidos

Autorización de Difusión.....	5
Agradecimientos.....	9
Resumen	11
Abstract.....	12
Índices de contenidos.....	14
Capítulo 1 - Introducción.....	16
1.1 Objetivos.....	21
Objetivo general	21
Objetivos específicos	22
1.2 Metodología.....	22
1.3 Organización de la memoria.....	23
Capítulo 2 - Estado del arte	24
2.1 Revisión de trabajos.....	25
2.2 Problemática y solución propuesta	32
Capítulo 3 -Marco teórico.....	34
3.1 Formación de imágenes digitales y diseño de la plataforma	35
Diseño de la plataforma	36
Fundamentos del color.....	37
Segmentación.....	42
3.2 Operaciones Morfológicas.....	43
Binzarización	46
Apertura – Erosión y Dilatación	47
3.3 Etiquetado de componentes conexas	48
Vencindad entre píxeles.....	48
Conectividad entre píxeles.....	49
Etiquetado de regiones.....	51
3.4 Identificación de regiones.....	52
Propiedades métricas	52
Propiedades topológicas	55

Momentos geométricos.....	56
Chapter 4 - Modelo de identificación: Diseño del proyecto.....	59
Chapter 5 - Resultados.....	65
5.1 Ejemplo de imágenes.....	65
5.2 Análisis de imágenes	75
5.2 Tiempo de computación	77
Chapter 6 - Conclusión y líneas futuras.....	79
Referencias	83
Chapter 7 - Introduction	89
Chapter 8 - Conclusion and future lines	94

Capítulo 1 - Introducción

Es bien sabido cómo el uso de imágenes tiene múltiples aplicaciones, en áreas tan diversas como la medicina, la agricultura, la industria en general y la industria 4.0 en particular, y en estos últimos tiempos en el ámbito de los sistemas conectados donde los avances son ciertamente significativos.

Dentro de los amplios campos de aplicación, cabe centrarse en uno muy específico de futuro inmediato. Como bien es sabido, los avances en el ámbito de los vehículos aéreos no tripulados (*Unmanned Aerial Vehicles, UAV*) están llamados a jugar un papel importante en multitud de áreas de aplicación y diversos tópicos de aplicación (Pajares 2015): agricultura, silvicultura, monitorización y exploración de catástrofes producidas por fuego, terremotos, huracanes, tsunamis, avalanchas, escapes nucleares entre otros o para localización, rescate, realización de inventarios forestales y de animales, vigilancia con fines de seguridad, control del tráfico rodado o fotogrametría entre otras muchas posibles aplicaciones, incluyendo incluso la entrega de paquetería por determinadas compañías de distribución de productos.

Los UAVs, están equipados con diversas tecnologías sensoriales tanto para navegación (GPS, Unidades de Medida Inercial, Sistemas de comunicación, altímetros, giróscopos) como para la detección en función de la misión que tiene asignada. En este último caso destacan por ejemplo los sistemas LIDAR (Laser Imaging Detection and Ranging), Radar y especialmente cámaras operando en diferentes rangos del espectro. Cabe destacar en este sentido las de naturaleza infra-roja y dentro de éstas las térmicas, las multiespectrales, que capturan radiaciones en una amplia gama de longitudes de onda y específicamente las cámaras operando en el rango visible del espectro, comúnmente conocidas como cámaras RGB.

En la figura 1.1 se muestran una secuencia de vuelo de un UAV, en este caso un cuatri-rotor, durante diferentes pruebas realizadas al amparo del proyecto SALACOM (2013), que ha dado origen al presente trabajo. El UAV está equipado con un *Smartphone*

Galaxy S6 para captura de imágenes con la cámara de este dispositivo, cuya resolución es de 1440×2560 píxeles y bajo el sistema operativo Android. En el mencionado proyecto SALACOM se desarrollaron sendos sistemas autónomos cooperativos y colaborativos. El UAV tiene la misión de explorar el entorno de operación para detectar posibles zonas de vertidos contaminantes sobre la superficie del agua mediante la captura de imágenes, para proporcionar dicha información al barco autónomo que ha de dirigirse a la zona del vertido con fines de contención y operación sobre él para evitar y en todo caso minimizar el posible impacto medioambiental posible. Todo ello bajo el diseño y planteamiento de un sistema cooperativo entre vehículos autónomos.

Figura 1.1 Una secuencia de vuelo de un cuatri-rotor equipado con un teléfono móvil.



Como bien es sabido la autonomía de cualquier UAV es limitada y en cualquier caso, aun estando todavía operativo, al finalizar la misión tanto en el ámbito del proyecto SALACOM como en cualquier tipo de operación el UAV debe posarse en un sitio seguro. En el proyecto SALACOM el aterrizaje debe realizarse sobre una plataforma específicamente diseñada y patentada a tal efecto (Cruz y col., 2012). La citada plataforma se muestra en la figura 1.2. En (a) vista desde arriba y en (b) con el UAV posado sobre ella. Mediante el sistema de visión a bordo del propio UAV se ha diseñado un procedimiento de identificación para su reconocimiento basado en el procesamiento de imágenes y la aplicación de técnicas específicas de Visión por Computador, siempre bajo la consideración de que estos métodos deben operar en entornos de exterior donde las condiciones ambientales, principalmente derivadas de la iluminación incontrolada, son completamente adversas, lo que dificulta enormemente su aplicación de cara a conseguir resultados satisfactorios.

Figura 1.2 Vistas de la plataforma de aterrizaje patentada.



(A)



(B)

En el ámbito del proyecto SALACOM el aterrizaje ha de realizarse sobre el barco autónomo, que lleva la plataforma a bordo para el posado mediante su reconocimiento a través del sistema de visión, figura 1.3. Obviamente, en este caso, se trata de una situación crítica el hecho de que el aterrizaje se produzca de forma segura y precisa sobre ella, ya que de lo contrario el UAV caería sobre el agua con su consiguiente deterioro y posible pérdida definitiva. En cualquier caso, el aterrizaje debe realizarse siempre de forma segura para evitar que éste se produzca en zonas críticas que pongan en peligro el propio UAV y a veces la propia misión.

Figura 1.3 Plataforma de aterrizaje de UAVs a bordo de un barco autónomo.



No obstante, como se verá posteriormente en relación con el estado del arte, este diseño entraña dificultades intrínsecas respecto de su reconocimiento mediante las técnicas de visión por computador propuestas. En efecto, dado que el UAV opera en entornos de exterior, las condiciones de iluminación son altamente cambiantes, apareciendo brillos o cambios bruscos de tonalidades que llevan a reconocimientos erróneos o incluso a la falta de reconocimiento. De hecho, en la propia figura 1.2 puede verse cómo la tonalidad negra en (a) cambia a azul en (b). Además, debido a esos cambios adversos de iluminación y brillos, a veces determinadas zonas negras u oscuras aparecen blancas, por lo que el reconocimiento llega a ser complicado. Esta es la principal razón

que motiva el presente trabajo, cuya finalidad es el diseño de una plataforma dotada de la mayor robustez posible, desde el punto de vista de su reconocimiento mediante la aplicación de técnicas específicas de procesamiento de imágenes para su posterior reconocimiento en el ámbito de la Visión por Computador.

En Visión por Computador cabe distinguir conceptualmente entre lo que se conoce como visión de Alto y Bajo nivel. Siguiendo los planteamientos expresados en Pajares y Cruz (2007*a,b*) o Davies (2018) la visión de bajo nivel hace referencia a los procesos inherentes realizados en cualquier aplicación desde la captura de imágenes hasta la interpretación de la escena, que corresponde ya al proceso de alto nivel.

Más específicamente, en lo concerniente al bajo nivel se comienza con la captura de imágenes con cámaras digitales según la aplicación (Pajares y col., 2016). Los dispositivos sensores son de carácter digital, obteniendo las matrices de valores de intensidad cuantizados según el sensor y la banda espectral a la que dichos dispositivos son sensibles. En el presente proyecto se trata de imágenes en el espectro visible y por tanto RGB. Estas imágenes se pre-procesan para mejorar su calidad, entre ellas se realizan operaciones para eliminar el ruido, corregir problemas causados por la iluminación y los tiempos de exposición, entre las que se puede citar el realzado de imágenes. Todas ellas permiten llevar a cabo operaciones conocidas como segmentación de imágenes, cuya finalidad es la identificación y determinación de objetos en la escena (González y Woods, 2008; Pajares y Cruz, 2007*a*; Davies, 2018).

En este trabajo se aborda el problema de reconocimiento de figuras sobre una base homogénea (fondo blanco), que es en definitiva la plataforma objeto de reconocimiento. Su diseño se explica posteriormente. Este problema es abordado en las etapas de segmentación, identificación de patrones asociadas a las formas de los objetos y reconocimiento de figuras una vez considerados dichos patrones. En la primera etapa, se aplican técnicas de extracción de píxeles en base a sus colores fundamentales (RGB), que permite identificar las figuras sobre el fondo blanco. Las áreas identificadas se etiquetan convenientemente, con el fin de extraer propiedades

relevantes de las figuras contenidas sobre la base. Dichas propiedades se determinan en función de las características geométricas que las definen. El proceso de reconocimiento de la plataforma se establece mediante un procedimiento de decisión basado en la información de color y sus propiedades geométricas. El diseño de esta plataforma se establece en base a la problemática analizada en el estado del arte, con el fin de proponer un diseño de plataforma acompañado de un método de visión por computador, con técnicas de alto y bajo nivel, de aquí en adelante identificado como algoritmo.

En la etapa de pruebas se verifica la robustez de la propuesta en entornos de exterior en los que la iluminación constituye un punto crítico. La primera parte del algoritmo detecta las figuras en entorno de baja iluminación. Sin embargo, en entornos agresivos presenta limitaciones, que son corregidas por el segundo componente del algoritmo.

En el siguiente enlace se puede acceder a todo el repositorio donde se incluye todo el código del proyecto, así como, una imagen que sirve como ejemplo. El enlace es:
https://drive.google.com/drive/folders/1wZuAiLkdIyj4HJ9Pda9h_ZxWxOowHSTC?usp=sharing

1.1 Objetivos

En este apartado, se presentan los objetivos del trabajo, que se han desagregado en objetivos de tipo general y específicos.

Objetivo general

El objetivo general del trabajo final consiste en desarrollar un método de reconocimiento de figuras específicas, formando parte de una plataforma para aterrizaje de UAVs, mediante técnicas de procesamiento digital de imágenes y visión por computador.

Las técnicas deben ser lo suficientemente robustas, para solventar la problemática derivada de su aplicación en entornos de exterior, donde las condiciones, principalmente de iluminación son ciertamente adversas.

Objetivos específicos

Los objetivos específicos del procesamiento de imágenes son los siguientes:

- Realizar el preprocesamiento de las imágenes para reducir ruidos y corregir la iluminación.
- Llevar a cabo la determinación de las imágenes mediante la extracción de píxeles en base a sus colores fundamentales RGB.
- Realizar la segmentación de las imágenes para determinar el contorno de las figuras contenidas en las imágenes.
- Determinar las figuras por la forma de los objetos mediante técnicas de momentos de Hu.

1.2 Metodología

Para el desarrollo del trabajo se han seguido las pautas establecidas en el método científico. Como primer paso se ha realizado una búsqueda bibliográfica en todas las fuentes académicas disponibles, utilizando bases de datos internacionales a las cuales la Universidad Complutense de Madrid tiene acceso, así como otras fuentes de carácter público. Tras lo cual se plantean las etapas de trabajo que se indican a continuación:

- Definición de la problemática que se desea abordar mediante el uso de métodos de reconocimiento de figuras sobre una base dada en entornos de exterior.
- Planteamiento del alcance y objetivos del trabajo
- Diseño y desarrollo de las metodologías identificadas.
- Realización de experimentos y obtención de resultados experimentales.
- Análisis e interpretación de los datos.
- Conclusión general.

- Planteamiento de trabajo futuro.

El proceso completo incluye la adquisición de datos, procesamiento de la información, desarrollo y codificación de las técnicas seleccionadas, obtención de resultados y finalmente su validación.

1.3 Organización de la memoria

La presente memoria se organiza de forma que en el capítulo dos se realiza una revisión sobre el estado del arte relativo a métodos de reconocimiento de plataformas para el aterrizaje de UAVs. El capítulo tres describe el marco teórico relativo a las técnicas y métodos de procesamiento de imágenes y visión por computador aplicadas. En el capítulo cuatro, se describe el diseño de la aplicación en su conjunto. El capítulo cinco recoge los experimentos realizados, junto con las consideraciones pertinentes al respecto. Finalmente, en el capítulo seis se plantean las conclusiones generales y las líneas de actuación futura.

Capítulo 2 - Estado del arte

Existen en la literatura una amplia variedad de métodos para el reconocimiento de plataformas de aterrizaje utilizando métodos basados en visión por computador, incluyendo los denominados activos y pasivos, para el aterrizaje de UAVs ([Kong et al. 2017](#); [Nguyen et al. 2017](#)). Los métodos activos contienen información adicional para ayuda y guía del UAV durante el aterrizaje tal como lámparas ([Anitha and Kumar, 2012](#)) o emisores de infrarojos (IR) ([Xu et al. 2009, 2013](#)). Incluso existen UAVs equipados con emisores IR instalados en la parte frontal, cuya radiación es captada por un sistema de cámaras en tierra, que guían el aterrizaje ([Yang et al. 2016](#)). También se han propuesto sistemas IR basados en visión estereoscópica ([Kong et al. 2013](#); [Zhou et al. 2015](#); [Tang et al. 2016](#)), en algunos otros casos se instalan emisores LEDs en el terreno para ser localizados con el sistema de visión a bordo ([Gui et al. 2013](#)). Por otra parte, [Maturana y Scherer \(2015\)](#) utilizan técnicas basadas en LIDAR para determinar la estructura 3D de la escena y a partir de ahí decidir la zona más apropiada para el aterrizaje si es que existe.

En contraposición a los sistemas activos, los pasivos solo requieren la visión por computador como sistema para el aterrizaje. Algunas propuestas tratan de reconocer el UAV en vuelo, para proceder a su guiado durante el aterrizaje con ayuda de una estación de control en tierra ([Martínez et al. 2009](#); [Kong et al. 2014](#)). Otras estrategias identifican un área segura del terreno mediante análisis de textura de éste, utilizando métodos de aprendizaje automático ([Li, 2013](#)), o mediante reconstrucción 3D del terreno para aterrizar en zonas planas ([Forster et al. 2015](#); [Johnson et al. 2005](#); [Bosch et al. 2006](#); [Desaraju et al. 2015](#)), o incluso utilizando técnicas de posicionamiento 3D con dos cámaras situadas en el suelo para UAVs de ala fija ([Kong et al. 2017](#)). Otra práctica común es la colocación de marcadores en el suelo y por tanto, utilizando plataformas de aprendizaje con diseños específicos para ser reconocidos mediante sistemas de visión. Algunos de estos marcadores a veces se sitúan a bordo de vehículos terrestres en movimiento sobre los que se posa el UAV ([Davide et al. 2017](#); [Lee et al. 2012, 2016](#)) o en barcos ([Polvara et al. 2017a](#); [Lin et al. 2017](#)).

Seguidamente se realiza una revisión sobre diferentes plataformas que plantean una serie de trabajos relacionados, que sirven de base para el planteamiento de la problemática y la solución propuesta.

2.1 Revisión de trabajos

[García-Pulido et al. \(2017\)](#) diseñaron un sistema experto que proporciona la orientación del UAV respecto de la plataforma junto con el grado de confianza en el reconocimiento de la plataforma. La plataforma utilizada es la mostrada en la figura 2.1 cuya patente es la descrita en [Cruz y col. \(2012\)](#). Se trata de una figura geométrica sobre fondo blanco conteniendo una serie de figuras geométricas inexistentes o al menos poco comunes en la naturaleza. El reconocimiento de la misma se realiza mediante descriptores de formas que permiten identificar las figuras de que consta. Básicamente los descriptores utilizados son el área, la excentricidad y la distancia relativa entre figuras, los cuales convenientemente combinados permiten establecer relaciones geométricas entre las formas de las diferentes figuras. En cuanto a las técnicas de procesamiento de imágenes se aplica umbralización seguida de binarización para la segmentación de las figuras. La identificación se basa en el método que se propone en dicho trabajo denominado Análisis Parcial Comparativo que proporciona el mencionado grado de confianza. La figura 4 muestra un ejemplo de visualización de dicha plataforma, junto con el resultado de la binarización y el etiquetado de las regiones sobre la misma.

La ausencia de color junto con el uso de figuras geométricas complejas y las métricas utilizadas hacen que en ocasiones no se reconozca la plataforma por falta de visualización de alguna o varias de las figuras. Esto ocurre por distorsiones de las figuras en las imágenes debido a la formación de éstas debido a la proyección de perspectiva que hace que círculos aparezcan proyectados como elipses. También surgen problemas serios debidos a la saturación de intensidad como consecuencia de la iluminación, tanto directa como indirecta, procedentes del sol, originando que partes

oscuras (negras) aparezcan en blanco debido a los reflejos producidos. Esto hace que determinadas figuras de la imagen desaparezcan, sin posibilidad de poder llegar a reconocerla en su conjunto, ya que la necesaria inter-relación entre las diferentes figuras no puede establecerse correctamente.

Figura 2.1 Plataforma de aterrizaje original, binarizada y etiquetada en regiones.



[Chen et al. \(2017\)](#) proponen una plataforma de aterrizaje con una figura en blanco y negro que contiene un pentágono en negro situado en el centro de un círculo que sirve como indicador del punto exacto de aterrizaje. El método para su reconocimiento se basa en la técnica conocida como *Faster Regions with Convolutional* (Faster R-CNN), incluyendo las típicas arquitecturas de las redes neuronales convolucionales con sus capas de convolución, *pooling* o *relu*. Este patrón es fácil de reconocer con relativamente baja complejidad y buen desempeño computacional. [Polvara et al. \(2017b\)](#) también aplicaron CNNs para detectar un círculo negro rodeando una cruz también negra, ambos sobre un fondo blanco. No obstante, a pesar de esto no se proporcionan mecanismos adicionales en ambos diseños para llevar a cabo el reconocimiento en el caso de que falten partes de las figuras que componen la plataforma debido a saturaciones derivadas de brillos o iluminación intensa. Por otra parte, la búsqueda se realiza de forma exhaustiva sobre la imagen sin tener en cuenta criterios de búsqueda específicos, como es el diseño de la plataforma, en particular en lo que se refiere a las componentes espectrales y las formas de las figuras que las definen.

Wang et al. (2016) propusieron una figura en forma de H, como la existente en los helipuertos convencionales, encerrada en un círculo, ambas en negro sobre fondo blanco o viceversa. La imagen se binariza mediante técnicas de umbralización para su posterior binarización con el objetivo de obtener puntos de interés, exactamente esquinas (*corners*), estableciendo una relación geométrica entre dichos puntos. Este tipo de planteamiento se ve afectado por la proyección de perspectiva, que entre otras cosas los círculos se proyectan como elipses debido a la distorsión del plano donde se encuentra la plataforma. De nuevo se pone de manifiesto que el uso de figuras simples sobre las que se establecen relaciones geométricas no resulta del todo eficiente, surgiendo problemas tales como ausencia de determinadas partes debido a los consabidos problemas de saturación en los niveles de intensidad. Por otra parte este tipo de planteamientos solo sirven cuando la plataforma aparece íntegramente en la imagen total (algo que aparece con frecuencia cuando la altitud del UAV es relativamente baja), no siendo viables en los casos en los que aparece parcialmente, bien como consecuencia de la captura o por problemas derivados de la iluminación. Sharp et al. (2001) también utilizaron cuadrados blancos y negros con detección de *corners* con el mismo propósito y problemática asociada. Zhao and Pei (2013) utilizaron una forma en H de color verde sobre la que aplicaron el bien conocido descriptor SURF (Speeded-Up Robust Features) de puntos de interés, el cual es invariante a pequeñas rotaciones y cambios de escala. En cualquier caso, el problema de ausencia de determinadas partes persiste con consecuencias negativas. Además, el color verde es relativamente abundante en entornos naturales, lo que dificulta su detección y finalmente, el hecho de que este detector sólo sea invariante a relativamente pequeñas rotaciones hace que la identificación de la plataforma sólo sea posible bajo determinados puntos de vista y orientación del UAV. Saripalli et al. (2006, 2012) también utilizaron una forma H, sobre la que aplicaron las siguientes técnicas de visión por computador: filtrado, umbralización, segmentación y etiquetado con determinación del ángulo de orientación entre las regiones etiquetadas con respecto al UAV. Aquí aparece la misma problemática en relación a la ausencia de determinadas partes y a que no es posible obtener una orientación apropiada en todo el rango de 0° a

360°. [Lange et al. \(2008, 2009\)](#) propusieron una figura similar a la de Saripalli et al. (2012) conteniendo diversos anillos concéntricos blancos y utilizando razones entre las dimensiones de los anillos más internos y más externos, lo que permite preservar la invarianza a diferentes altitudes de vuelo. No obstante, el principal problema estriba en que se requiere un posicionamiento perpendicular del eje óptico de la cámara con respecto a la plataforma para evitar distorsiones de los anillos circulares debido a la proyección de perspectiva.

[Cocchioni et al. \(2014\)](#) diseñaron sendos círculos concéntricos junto con dos triángulos equiláteros más pequeños, éstos de diferentes dimensiones. [Li et al. \(2012\)](#) también utilizaron un patrón basado en círculos concéntricos, concretamente seis convenientemente combinados. El algoritmo de reconocimiento se fundamenta en la aplicación de momentos invariantes afines, en conjunción con un clasificador basado en máquinas de vectores soporte (SVM, *Support Vector Machines*). De nuevo las distorsiones debidas a la proyección de perspectiva junto con el hecho de que las figuras no contienen color, conlleva asociada la problemática indicada previamente respecto de reflejos originando que determinadas zonas de la plataforma dejen de visualizarse.

[Guili et al. \(2009\)](#) diseñaron una plataforma con forma de T, cuyo material produce distintos niveles de intensidad en el infra-rojo con respecto al entorno, por lo que su visualización es posible con cámaras operando en el infra-rojo. Utilizaron las siguientes técnicas de visión por computador: extracción de bordes, momentos invariantes para su reconocimiento, y segmentación de la T. Esto requiere el uso de una cámara térmica. La ventaja ciertamente estriba en el hecho de que se puede utilizar en condiciones de nocturnidad.

Las bien conocidas AprilTags ([Olson, 2011](#); [AprilTag, 2018](#)) se han utilizado en [Ling \(2014\)](#); [Kyristis et al. \(2016\)](#); [Garrido-Jurado et al. \(2014\)](#), y se han aplicado detectores de bordes, con el fin de afrontar bajas resoluciones de la imagen además de las oclusiones a la vez que se asumen variaciones en cuanto a la intensidad y cambios

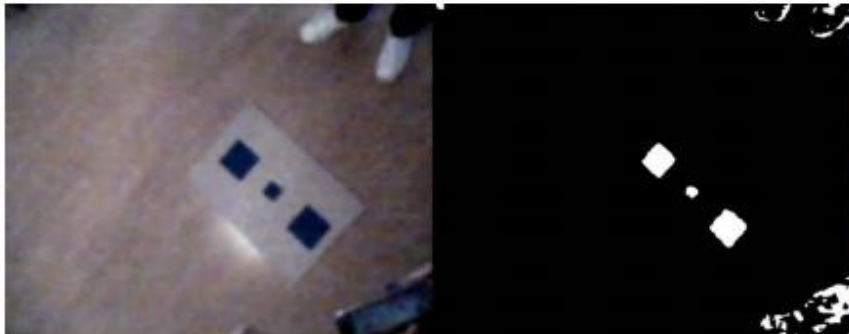
en la iluminación. Marcadores del tipo definido en [ArUco Markers \(2018\)](#) se han utilizado en [Chaves et al. \(2015\)](#) junto con el filtro de Kalman para guiar al UAV durante la última fase del aterrizaje sobre barcos. Sin embargo, resulta bien conocido el hecho de que los extractores de bordes son muy sensibles al ruido, generando bordes con rupturas, llevando a detecciones incorrectas. [Araar et al. \(2017\)](#) utilizaron también los AprilTags y aplicaron técnicas de agrupamiento por las magnitudes y direcciones del gradiente. Debido a que estos marcadores se componen de cuadrados se vuelve a reproducir el problema de las distorsiones por la proyección de perspectiva.

[Nguyen et al. \(2017\)](#) utilizaron tres círculos concéntricos que definen ocho áreas blancas y negras. La figura puede reconocerse incluso aunque falten áreas. Por el contrario, requiere determinar un punto de comienzo, lo que representa un serio inconveniente. Existe un área asimétrica que define la orientación de la plataforma con respecto al UAV. Se utiliza una plantilla para establecer la correspondencia entre ésta y la plataforma que se busca. Para el proceso de búsqueda se utiliza también el filtro de Kalman. El centro de la plataforma se identifica mediante la técnica de umbral adaptativo (Gonzalez y Woods, 2008). Cuando se identifica dicho punto, se definen puntos de intersección como transiciones de negro a blanco siguiendo el círculo que marca el centro. Lo indicado previamente sobre el uso de figuras en blanco y negro con respecto a brillos e iluminación es aplicable aquí. Por otra parte, la técnica de umbral adaptativo tiende a favorecer la extracción de bordes frente a regiones y además la definición de plantillas resulta complicado dadas las variaciones de perspectiva e inclinación del UAV con respecto a la plataforma.

[Gil-Fernández et al. \(2016\)](#) Desarrolló un software que permite identificar 3 rectángulos alineados de forma horizontal de color azul (Figura 2.2). Con ese patrón se aplica un algoritmo de segmentación de color, con el objetivo de crear una imagen binaria en blanco y negro (Figura 4-b), donde el blanco destacará el relleno azul de los rectángulos. Esta imagen binaria se tratará para identificar las figuras que coincidan con la plataforma reconociendo el tamaño y distancia entre las figuras. Cuando se haya

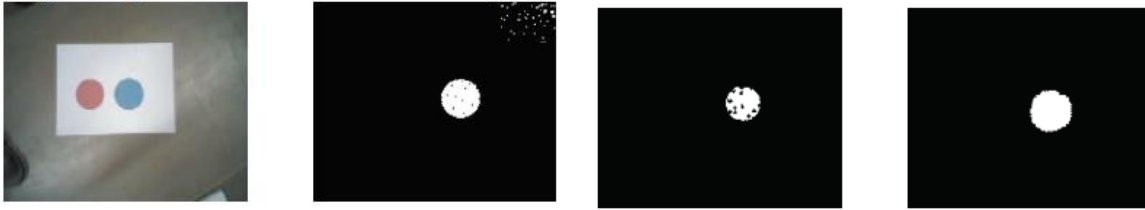
realizado este proceso de manera satisfactoria, el centro geométrico del rectángulo del medio se considerará como punto de aterrizaje.

Figura 2.2 Plataforma con tres rectángulos.



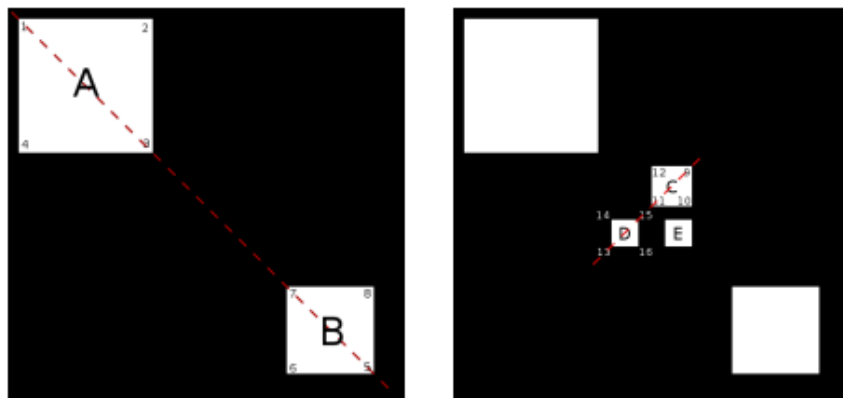
[Barták et al. \(2014\)](#) utilizan como plataforma un patrón con dos círculos de colores para ser reconocidos por la detección de manchas. El usuario puede usar cualquiera de los dos colores; solo es necesario especificar los rangos adecuados para el tono, saturación y el valor (HSV) de ambos círculos, para distinguirlos mediante el filtrado de paso de banda HSV. Con este proceso la imagen pasa a ser binaria (dos colores), lo que es apropiado para la detección de manchas. A continuación, y ante la posibilidad de que haya ruido en las imágenes, aplican el procedimiento de erosión, que consiste en tomar subáreas más grandes y usa el punto más oscuro del área para todos los puntos del área. Pero con este proceso ese ruido puede aumentar de tamaño por lo que por último aplican el proceso de dilatación que es similar al de erosión, pero utilizan esta vez el punto más claro para definir el color de la subárea, Figura 2.3.

Figura 2.3 Plataforma con círculos.



[Hermansson \(2010\)](#) diseña una plataforma muy parecida al ejemplo 8. En este caso también utilizan cuadrados, debido a que las esquinas son fácilmente detectables. Ellos modifican el algoritmo que detecta las esquinas calculando la distancia máxima desde el centro hasta sus bordes. Utiliza principalmente 2 cuadrados blancos, mientras que el fondo es negro para que contraste mejor. Un cuadrado más grande (A) y uno más pequeño (B) se colocan de manera que la diagonal de A coincida con la de B. Por lo tanto, cuatro puntos significativos siempre deben formar una línea recta, utilizándolo para validar la corrección de los puntos significativos encontrados. (Figura 2.4-a). Según la cámara se acerca a la plataforma, la distancia entre ella y el patrón disminuirá y no podrá ver los cuadrados. Para ello, necesita un patrón de menor escala, Este patrón se encuentra entre las figuras A y B, similar a esas figuras, pero girados 90° y para evitar confusión con las figuras grandes agregan un tercer cuadrado del mismo tamaño que el cuadrado más pequeño, Figura 2.4.

Figura 2.4 Plataforma con cuadrados y subcuadrados.



En definitiva, algunos métodos ([Chen et al. 2017](#); [Wang et al. 2016](#); [Saripalli et al. 2012](#); [Guili et al. 2009](#)) utilizan una única figura que no soluciona el problema de la ausencia de regiones debido a la saturación por exceso de brillo debido a la iluminación directa que origina abundantes reflexiones. Otros métodos ([García-Pulido et al. 2017](#); [Li et al. 2012](#); [Lange et al. 2008](#); [Ling, 2014](#); [Kyristsis et al. 2016](#); [Garrido-Jurado et al. 2014](#); [ArUco Markers, 2018](#)) utilizan varias figuras intercaladas para abordar el problema de las regiones que faltan, o figuras complejas basadas en círculos y elipses. Algunas de ellas ([García-Pulido et al. 2017](#)) permiten incluso una cierta distorsión en el plano debido a la inclinación de la cámara y a la proyección de perspectiva, si bien, otras propuestas como [Lange et al. \(2008\)](#), asumen que el UAV se debe situar paralelo al suelo durante el aterrizaje, ignorando el ángulo de inclinación. Los bordes y los puntos de interés son características consideradas como de alta sensibilidad al ruido ([Wang et al. 2016](#); [Sharp et al. 2001](#); [Zhao and Pei, 2013](#); [Guili et al. 2009](#)).

2.2 Problemática y solución propuesta

Los proyectos anteriormente analizados cumplen satisfactoriamente con el objetivo de detectar una plataforma mediante computación y de forma autónoma para el aterrizaje de un UAV, Pero en su mayoría deben detectar todas las figuras o figura de la plataforma para determinar que ahí se encuentra, lo cual limita mucho la posibilidad de acierto, ya que, si por alguna razón no se realiza una buena captura de la plataforma que la impida verse con claridad, la detección de esta resultará fallida.

Sin embargo, como se ha mencionado anteriormente, no todos los ejemplos requieren de una identificación total de la plataforma. Por ejemplo, el caso número 1 permite margen de error, debido a la implementación de un sistema de probabilidades con el que no es necesario que identifique toda la figura para determinar si es correcta o no. Aunque por su elevada proximidad, es fácil que, si por alguna razón una parte de su figura no sea distinguible, también sea igual para las demás.

Como solución, el proyecto descrito en este trabajo toma ideas del trabajo propuesto en [García-Pulido et al. \(2017\)](#) pero con algunas ampliaciones y modificaciones relativas a la inclusión del color. En vez de una única figura compuesta por subfiguras, se distribuyen cuatro figuras por las cuatro esquinas de la plataforma, que es concretamente un folio de tamaño A4. De esta forma si una o varias no se pueden identificar, todavía cabe identificar las restantes, pudiendo determinar así que se trata realmente de la plataforma. En efecto, al estar bastante más separadas, la probabilidad de que si una no se ve no se visualicen las otras, es más pequeñas que si se encuentran más próximas. Además, también las figuras tienen geometría y colores diferentes para añadir más probabilidades de acierto a la hora de distinguirlas.

Capítulo 3 -Marco teórico

El planteamiento que se formula tiene su fundamento en el tratamiento de imágenes en base al procesamiento de éstas según los conceptos que se exponen a lo largo del presente capítulo, tal y como se muestran esquemáticamente en la figura 3.1

Figura 3.1 Marco teórico conceptual



Para ello, en primer lugar, se establece el marco teórico sobre el concepto de imagen digital, que desemboca en el diseño de una plataforma compuesta de figuras geométricas a color, razón por la cual se introducen los aspectos relevantes relacionados con el color y su formación, junto con dos modelos fundamentales del mismo. A partir de la imagen a color es necesario realizar una serie de procesos encaminados a la identificación de los elementos de interés, lo que pasa por un proceso

conocido como segmentación de imágenes, que incluyen técnicas tales como reducción del ruido y umbralización. En este último caso con el propósito de aislar esos elementos de interés (en este caso la plataforma) respecto del fondo de la imagen. Tras este proceso se realizan las operaciones morfológicas pertinentes para mejorar la calidad de los resultados obtenidos en la fase de umbralización. El siguiente paso se encamina a la descripción de la imagen mediante sendos procesos que comprenden el etiquetado de las componentes conexas aplicando el concepto de conectividad en base a la vecindad espacial de los píxeles en la imagen. El paso final consiste en la identificación de las regiones, que en definitiva son las regiones que componen la figura de la plataforma. Dichos momentos tienen la propiedad intrínseca de que son invariantes a rotaciones, traslaciones y cambios de escala, circunstancia ésta que permite identificar dichas figuras independientemente de la distancia y perspectiva del dispositivo de captura con respecto a la plataforma, haciendo que las figuras aparezcan giradas, rotadas, trasladadas y con diferentes tamaños respecto del diseño original.

3.1 Formación de imágenes digitales y diseño de la plataforma

La unidad fundamental del procesamiento de imágenes se denomina píxel. Las imágenes se pueden representar mediante funciones bidimensionales, $f(x, y)$, en las cuales, x e y son coordenadas espaciales del plano. La amplitud de f en dichas coordenadas se denomina intensidad de la imagen en dicho punto. En las monocromáticas, esa intensidad determina el nivel de gris. También se puede definir a $f(x, y)$ como la representación del brillo en las coordenadas x e y .

El dominio que permite la representación de las imágenes se conoce como dominio espacial, que es similar a la representación de las señales discretas en el dominio del tiempo, de esta manera $f(x, y)$ se puede representar como una superficie, que representa distintos niveles de intensidad. En las imágenes monocromáticas, y con una representación de los niveles de intensidad de ocho bits, la máxima intensidad que corresponde al color blanco está representada por el valor máximo de esta

representación, esto es 255, mientras que el mínimo valor de intensidad se corresponde con el negro, que en la representación de 8 bits resulta ser el cero.

Las imágenes de color se representan mediante estructuras matriciales de dimensión $M \times N \times 3$, siendo M y N el ancho y alto respectivamente de las matrices, mientras que la tercera dimensión, en este caso con valor tres, se corresponde con tres componentes espectrales rojo, verde y azul (RGB), que constituyen el modelo de color conocido como RGB y donde cada una de estas matrices recibe la información espectral del canal correspondiente en el rango espectral asociado. En este modelo, cada píxel es un vector que requiere un almacenamiento de 24 bits, ya que cada componente es representado por 8 bits. Por otra parte, y en términos de representación de los niveles de intensidad, el tipo de datos que admite el sensor de captura determina el rango de sus valores, de esta forma, como se ha indicado previamente, en una representación de 8 bits el rango de valores corresponde al intervalo $[0, 255]$, con una representación de 16 bits el rango es $[0, 65535]$, mientras que para 32 bits es $[0, 4294967295]$. Las imágenes utilizadas en el presente proyecto poseen una representación de 8 bits, por canal, y, por tanto, al tratarse de imágenes con los tres canales RGB con la representación de 24 bits indicada previamente.

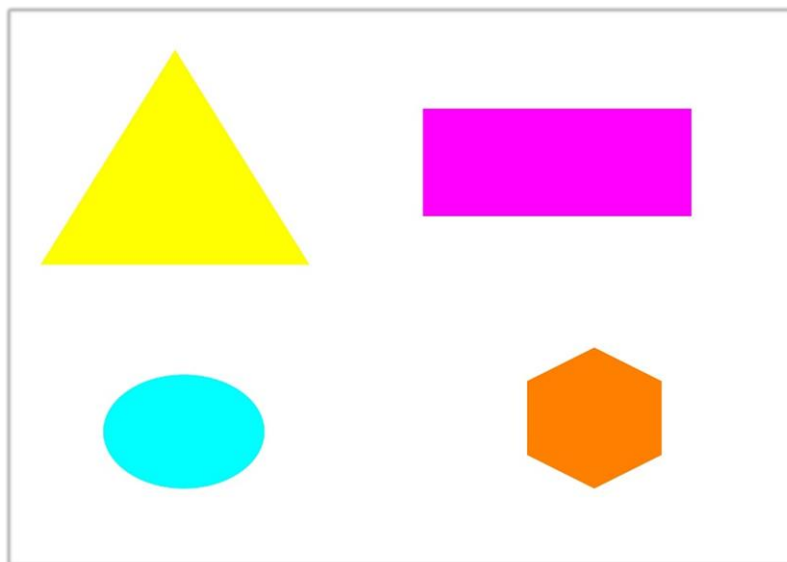
Por otra parte, como se verá posteriormente, tras el proceso de binarización los niveles de intensidad quedan reducidos a dos valores, que en representación de 8 bits se corresponden exactamente con el 0 (negro) y el 255 (blanco), traducándose posteriormente a lo que se conoce como una imagen lógica, con valores “0” y “1” respectivamente, que por otra parte constituyen la base conceptual de los procesamiento morfológicos y de etiquetado.

Diseño de la plataforma

La plataforma, figura 3.2, está compuesta por 4 figuras geométricas (rectángulo, triángulo, elipse y pentágono), dispuestas en una base de papel de tamaño A4 y rellenas con colores diferentes (magenta, rojo, amarillo y cian). La distancia para cada par de

figuras en vertical y horizontal con respecto a sus centroides, son 10 cm y 15 cm respectivamente. Este diseño refleja el objetivo propuesto, por un lado, disponer de colores poco frecuentes en la naturaleza o entornos urbanos, donde debe operar el UAV, lo que permitirá realizar la segmentación de una forma más eficiente. Además, contiene figuras cuyas formas geométricas son diferentes, cuya finalidad es poder discriminar entre ellas de una forma más eficiente mediante los descriptores de regiones basados en los momentos invariantes de Hu (Pajares y Cruz, 2007a, b; González y col., 2008).

Figura 3.2 Nuestra plataforma



Fundamentos del color

Resulta muy conocido el hecho de que la luz solar que atraviesa un prisma de cristal se descompone en los siete colores básicos del espectro visible. Básicamente, los colores que las personas perciben en un objeto están determinados por la naturaleza de la luz reflejada por el objeto. Un cuerpo que refleja la luz de todas las longitudes de onda se

muestra como blanco al observador. Sin embargo, un cuerpo que favorece la reflectancia en un rango limitado de longitudes de onda en el espectro visible exhibe un determinado color. Por ejemplo, los objetos verdes reflejan la luz con longitudes de onda en el rango de 500 a 570 nm y absorben gran cantidad de energía en otras longitudes de onda.

La caracterización de la luz es vital para la ciencia del color como se describe en el apéndice A. Desde el punto de vista del ojo humano, todos los colores son vistos como combinaciones variables de los tres colores primarios: rojo (R), verde (G) y azul (B). Con el fin de su estandarización, la CIE (Comisión Internationale de l'Eclairage) asignó en 1931 los siguientes valores de longitudes de onda a los colores primarios: azul=435,8 nm, verde=546,1 nm, y rojo=700 nm. Los colores primarios pueden mezclarse para producir los colores secundarios de la luz: magenta (rojo y azul), cyan (verde y azul) y amarillo (rojo y verde). La mezcla de los tres colores primarios o secundarios en proporciones iguales produce color blanco. Conviene diferenciar entre los colores primarios de la luz y de los pigmentos o colorantes. En lo que sigue, un color primario se define como aquel que absorbe un color primario de la luz y refleja o transmite los otros dos. Por tanto, los colores primarios de los pigmentos son magenta, cyan y amarillo y los secundarios rojo, verde y azul.

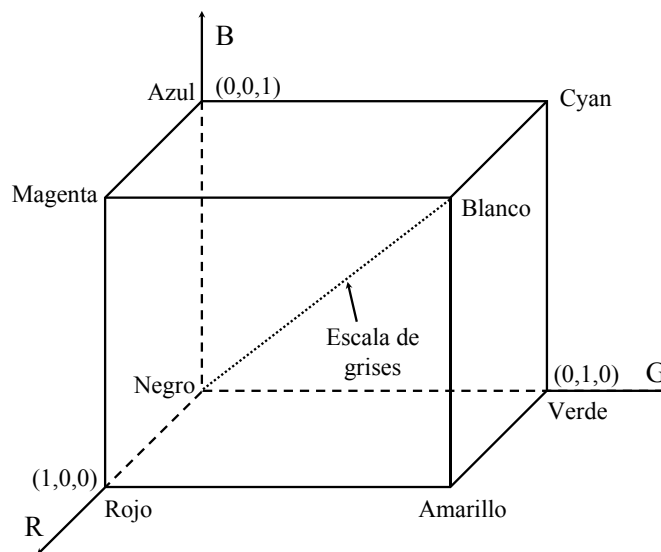
Desde el punto de vista de la aplicación del concepto de color en el desarrollo del presente proyecto son dos los modelos a considerar RGB, ya considerado y HSV (Hue, Saturation, Intensity).

- **Modelo RGB**

En el modelo RGB cada color aparece en sus componentes espectrales primarias: rojo, verde, azul. Este modelo está basado en el sistema de coordenadas cartesianas. El subespacio de color de interés es el tetraedro mostrado en la figura 3.3. En el cual los valores RGB están en tres vértices; cyan, magenta y amarillo se sitúan en otros tres vértices, el negro corresponde al origen y el blanco en el vértice más alejado del origen.

En este modelo, la escala de grises se extiende desde el negro al blanco a lo largo de la diagonal que une esos dos puntos, y los colores son puntos dentro del tetraedro, definidos por vectores desde el origen. Por conveniencia, se asume que todos los vectores han sido normalizados, de modo que el tetraedro de la figura 3.3 es el tetraedro unitario, es decir, todos los valores de R, G y B se sitúan en el rango $[0,1]$. Las imágenes en este modelo se forman por la combinación en diferentes proporciones de cada uno de los colores primarios.

































Figura 3.3 Tetraedro de color RGB



En la figura 3.4, se muestran diferentes combinaciones de valores para los tres canales RGB. De acuerdo con la figura 5.2, el blanco representa un valor de 1, el negro 0, el

gris claro 0,86 y el gris oscuro 0,57. Inyectando los valores correspondientes en los canales R, G, B de las columnas 1 2 y 3, se obtiene el resultado mostrado en la cuarta columna.

Figura 3.4 Combinaciones de valores R, G, B para derivar un determinado color

R	G	B	Resultado
			
			
			
			
			
			
			
			

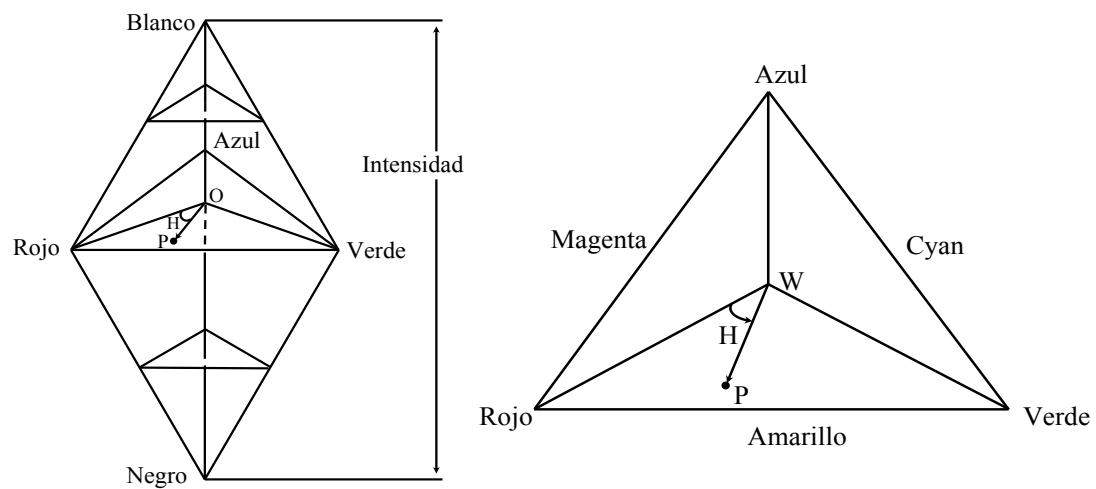
- **Modelos HSI y HSV**

El matiz es un atributo del color que describe su pureza (amarillo puro, rojo, naranja), mientras la saturación proporciona una medida del grado en el que el color puro es diluido con luz blanca. El modelo de color HSI debe su utilidad a dos hechos fundamentales. Primero la componente de intensidad I, se puede separar de la información del color en la imagen. Segundo las componentes de matiz y saturación están íntimamente relacionadas con el modo en que los humanos perciben el color. Estas características hacen del modelo HSI una herramienta ideal para desarrollar

algoritmos de procesamiento de imágenes basados en alguna de las sensaciones de color del sistema visual humano, siendo justamente este aspecto el considerado en el diseño y desarrollo de la plataforma de aterrizaje.

Algunos ejemplos de la utilidad de este modelo varían desde el diseño de sistemas para verificar el grado de madurez de las frutas o la inspección del acabado de color de determinados productos. La idea consiste en utilizar las propiedades del color del mismo modo que lo haría una persona encargada de esa misión, en este caso, el reconocimiento de la plataforma. El modelo HSV es similar al HSI salvo que la intensidad varía de negro a blanco en un único prisma. La figura 3.5 muestra la representación geométrica de ambos modelos. Siempre es posible la transformación entre los modelos RGB y HSI (HSV), tanto en un sentido como en el otro basándose en las relaciones geométricas mostradas en la mencionada figura 3.5, cuyos detalles se pueden encontrar en (Pajares y col., 2007a, b; González y col., 2008).

Figura 3.5 (a) Triángulo de color HSI; (b) Color e intensidad en el modelo HSI



(A)

(B)

Segmentación

La segmentación de imágenes es el primer paso para llevar a cabo el proceso de particionado de la imagen en regiones para luego poder analizar las propiedades de cada una por separado para poder concretar si la plataforma es correcta o no. Estas regiones comparten una serie de características como pueden ser la amplitud, los contornos o la textura. Para entender mejores procesos de este tipo vamos a comentar una serie técnicas y conceptos teóricos a continuación.

- **Umbralización**

Los niveles de gris de los píxeles que pertenecen al objeto son completamente diferentes a aquellos otros que pertenecen al fondo, en muchas aplicaciones de procesamiento de imágenes. La determinación del umbral se convierte en un proceso efectivo para separar esos objetos de primer plano del fondo. Podemos dividir los píxeles de la imagen en dos grupos principales, de acuerdo con su nivel de gris. Estos niveles de gris pueden servir como "detectores" para distinguir entre el fondo y los objetos que se considera como primer plano en la imagen. Selecciona un nivel de gris entre esos dos grupos principales de nivel de gris, que servirá como un umbral para distinguir los dos grupos (objetos y fondo). En el presente caso, el objeto es realmente la plataforma, mientras que el fondo lo constituye el resto de la escena.

- **Método Otsu**

El método de Otsu (1979), es un algoritmo de umbralización automático ampliamente utilizado con resultados satisfactorios. Realiza intuitivamente un umbral de imagen basado en agrupación, o la disminución de una imagen en escala de grises a una imagen binaria o de umbral. El algoritmo considera que la imagen contiene dos clases de píxeles que respaldan el histograma bimodal (píxeles de primer plano y píxeles de

fondo); luego enumera el umbral óptimo que desconecta las dos clases para que su dispersión combinada (varianza intra-clase) sea mínima.

Dada una imagen con L niveles de intensidad y asumiendo que el umbral buscado es T , las probabilidades acumuladas hasta T y desde T hasta L resulta ser en la ecuación 1,

$$w_1(t) = \sum_{z=1}^T P(z) \quad y \quad w_2(t) = \sum_{z=T+1}^L P(z) \quad (1)$$

A continuación, en las ecuaciones 2 y 3, se obtienen las medias y la varianza asociada,

$$\mu_1(t) = \sum_{z=1}^T zP(z) \quad y \quad \mu_2(t) = \sum_{z=T+1}^L zP(z) \quad (2)$$

$$\sigma_1^2(t) = \sum_{z=1}^T (z - \mu_1(t))^2 \frac{P(z)}{w_1(t)} \quad y \quad \sigma_2^2(t) = \sum_{z=T+1}^L (z - \mu_2(t))^2 \frac{P(z)}{w_2(t)} \quad (3)$$

Finalmente, en la ecuación 4, se obtiene la varianza ponderada

$$\sigma_w^2(t) = w_1(t)\sigma_1^2(t) + w_2(t)\sigma_2^2(t) \quad (4)$$

Dada una imagen con L niveles de intensidad y asumiendo que el umbral buscado es T , las probabilidades acumuladas hasta T y desde T hasta L resulta ser en la ecuación 1.

3.2 Operaciones Morfológicas

La morfología matemática aprovecha las propiedades de los conjuntos de puntos, los resultados de la geometría integral y la topología. La premisa inicial consiste en suponer que las imágenes reales pueden ser modeladas utilizando conjuntos de puntos de cualquier dimensión (por ejemplo, el espacio Euclídeo N-dimensional). El espacio Euclídeo 2D (E^2) y su sistema de subconjuntos es un dominio natural para la descripción de formas planas. Desde la perspectiva de conjuntos se consideran las operaciones habituales en ellos, a saber: inclusión (\subset), unión (\cup), intersección (\cap),

complementario (c) o conjunto vacío (\emptyset). La diferencia de conjuntos se define por $X - Y = X \cap Y^c$. Un punto se representa por un par de enteros que corresponden a las coordenadas de la imagen digital. Por tanto, una imagen binaria puede tratarse como un conjunto de puntos 2D. Los puntos que pertenecen a los objetos en la imagen representan un conjunto X . En imágenes binarias esos puntos son píxeles con valor binario uno. Los puntos del conjunto complementario X^c corresponden al fondo con valores binarios igual a cero. Fijándonos en la imagen de la figura 3.6, el origen, marcado con un punto, tiene coordenadas (0,0) y las coordenadas de cualquier punto se interpretan como (posición en la fila, posición en la columna) con respecto al origen. Además, en dicha figura, los puntos que pertenecen al objeto se marcan con “1”. Cualquier punto x de una imagen discreta X puede considerarse un vector con respecto al origen (0,0).

Figura 3.6 Ejemplo de conjunto de puntos

$$\begin{bmatrix} \bullet 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

En este caso el conjunto de puntos X considerando exclusivamente los valores lógicos 1 vendría dado por $X = \{(0,1), (0,2), (0,3), (1,1), (1,2), (1,3), (2,2), (2,3), (3,2), (3,3)\}$.

Una transformación morfológica viene dada por la relación de la imagen (conjunto de puntos X) con otro pequeño conjunto de puntos B , llamado elemento estructural. B se expresa con respecto a un origen local O (llamado punto representativo). Algunos elementos estructurales típicos (obviamente también desde el punto de vista binario) se muestran en la figura 3.7 en la cual el punto representativo se marca por un punto. El elemento estructural de esta figura tiene la siguiente expresión en forma de conjunto de puntos: $B = \{(-1, -1), (-1,0), (-1,1), (0, -1), (0,0), (0,1), (1, -1), (1,0), (1,1)\}$.

Figura 3.7 Elemento estructural

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & \bullet 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Binzarización

La binarización consiste en representar una imagen con dos tonos de color, normalmente blanco y negro. Para ello compara los niveles de grises que presenta una imagen con un umbral establecido. Si el nivel de gris es menor que el umbral calculado, se le asigna al píxel de la imagen binarizada el valor 0 (negro), y si es mayor, se le asigna un 1 (blanco).

En este proyecto es utilizada esta técnica principalmente para tratar de aislar las partes necesarias de la imagen, en este caso la plataforma a analizar. A continuación, en la figura 3.8, se presenta un ejemplo de binarización de nuestro proyecto.

Figura 3.8 Imagen original y binarizada



Apertura – Erosión y Dilatación

La transformación morfológica de la dilatación \oplus combina dos conjuntos utilizando la adición de vectores (o adición de conjuntos de Minkowski). Como se puede ver en la ecuación 5, la dilatación $X \oplus Y$ es el conjunto de puntos de todas las posibles adiciones vectoriales de pares de elementos, uno de cada conjunto X y B.

$$X \oplus B = \{d \in E^2 : d = x + b \text{ para cada } x \in X \text{ y } b \in B\} \quad (5)$$

De esta forma, la dilatación agrega una capa de píxeles a los límites interno y externo de las regiones segmentadas. Esta agregación se realiza por efecto del elemento estructural.

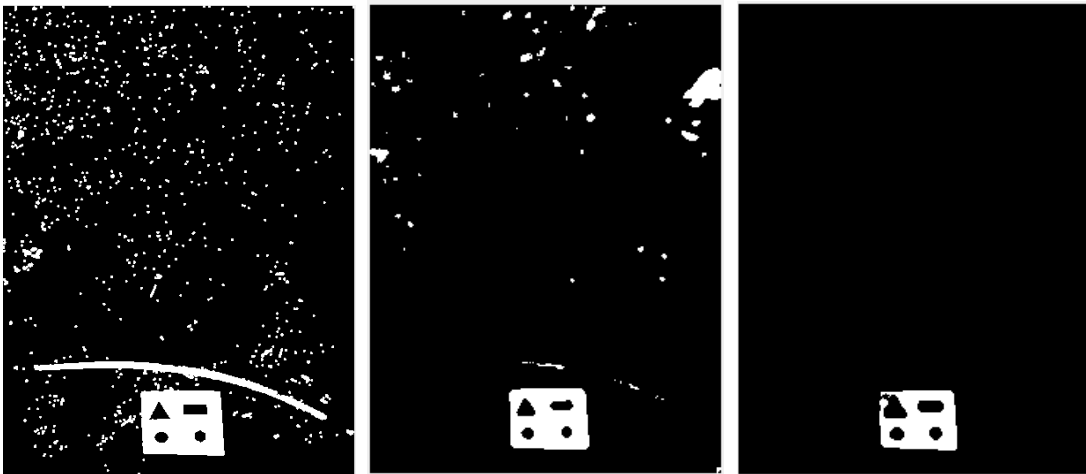
La transformación morfológica de la erosión \otimes combina dos conjuntos utilizando la substracción de vectores. Es dual de la dilatación. Ni la erosión ni la dilatación son transformaciones invertibles, puede verse en la ecuación 6.

$$X \otimes B = \{d \in E^2 : d = d + b \text{ para cada } b \in B\} \quad (6)$$

Una erosión seguida de una dilatación conforma lo que se denomina una operación de apertura, elegida en este proyecto, ya que de esta forma eliminaríamos las pequeñas regiones blancas.

En la figura 3.9(b) se muestra el resultado de la erosión sobre la imagen binarizada mostrada en (a), mientras en (c) se muestra el resultado de la dilatación sobre la imagen (b), que como se ha indicado previamente, se trata de una apertura.

Figura 3.9 Imágenes binarizadas (a), erosionada (b) y dilatada (c)



3.3 Etiquetado de componentes conexas

Para cada una de las regiones extraídas es necesario proceder a su identificación. Con tal propósito se procede a aplicar lo que se conoce como etiquetado de componentes conexas, entendiendo por tal el hecho de identificar una región con una etiqueta única. El concepto de región conexa se refiere al hecho de que todos los píxeles de la región pueden ser visitados sin abandonar la región. Para tal finalidad resulta de especial relevancia el concepto de vecindad entre píxeles. Ambos conceptos, vecindad y etiquetado se exponen a continuación.

Vecindad entre píxeles

Como se ha explicado en el apartado de binarización, en las imágenes digitales binarias, un píxel toma un valor lógico de 1 o 0 lógicos, representando respectivamente píxeles blancos y negros. Para identificar objetos en un patrón digital, necesitamos localizar

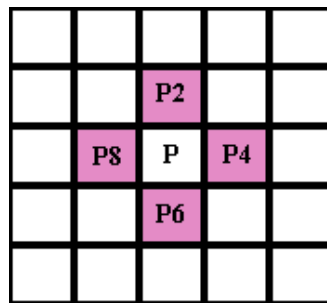
grupos de píxeles negros que estén "conectados" entre sí. En otras palabras, los objetos en un patrón digital dado son los componentes conectados de ese patrón. En general, un componente conectado es un conjunto de píxeles negros, P , de manera que para cada par de píxeles P_i y P_j en P , existe una secuencia de píxeles P_i, \dots, P_j tal que, todos los píxeles en la secuencia están en el conjunto P , es decir, son negros, y cada 2 píxeles que son adyacentes en la secuencia son "vecinos".

La solución para determinar que 2 píxeles son vecinos no es trivial con píxeles cuadrados, la razón es que, los píxeles comparten un borde, un vértice o ninguno de los dos. Hay 8 píxeles que comparten un borde o un vértice con cualquier píxel dado; estos píxeles forman el vecindario de Moore de ese píxel. Lo que da lugar a 2 tipos de conexiones: conectividad-4 y conectividad-8. [2]

Conectividad entre píxeles

Un píxel, Q , es un vecino 4 de un píxel dado, P , si Q y P comparten un borde. Los 4 vecinos del píxel P (es decir, los píxeles P_2 , P_4 , P_6 y P_8) se muestran en la figura 3.10.

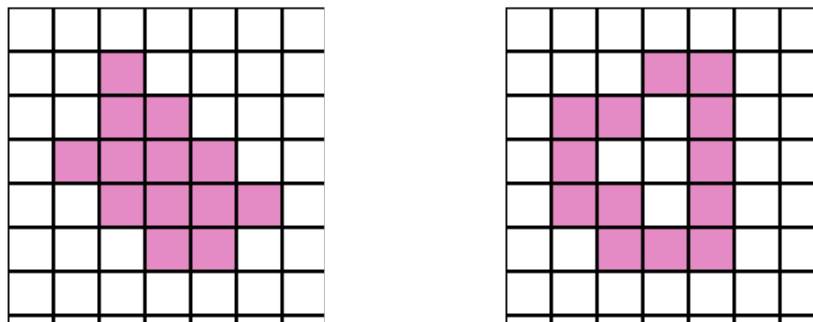
Figura 3.10 Vecindad-4



Por lo tanto la definición de un componte de conectividad-4 resulta ser: “un conjunto de píxeles negros, P , es un componente de conectividad-4, si para cada par de píxeles de píxeles P_i y P_j en P , existe una secuencia de píxeles P_i, \dots, P_j tal que, todos los píxeles de la secuencia están en el conjunto P , es decir, son negros, y cada 2 píxeles

adyacentes en la secuencia son 4 vecinos. Las rejillas mostradas en la figura 3.11 son dos ejemplos de patrones de conectividad-4.

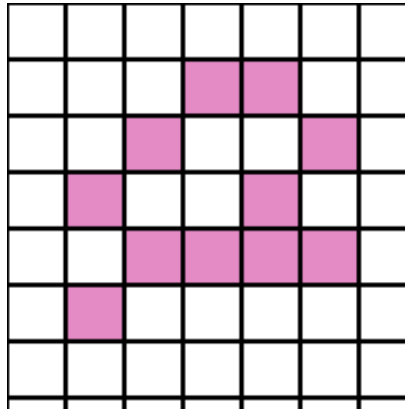
Figura 3.11 Patrón conectividad-4



Siguiendo el mismo criterio anterior, para la conectividad-8, se define el concepto de vecino 8 (vecino indirecto). Así, un píxel, Q, es un vecino 8 de un píxel dado, P, si Q y P comparten un borde o un vértice. Los 8 vecinos de un píxel P determinado forman el vecindario de Moore de ese píxel. Por lo tanto la definición de un componente de conectividad-8 es: “Un conjunto de píxeles negros, P, es un componente de conectividad-8, si para cada par de píxeles de píxeles P_i y P_j en P, existe una secuencia de píxeles P_i, \dots, P_j tal que, todos los píxeles de la secuencia están en el conjunto P, es decir, son negros, y cada 2 píxeles adyacentes en la secuencia son 8 vecinos”

Todos los patrones de conectividad-4 son de conectividad-8, es decir, los patrones de conectividad-4 son un subconjunto del conjunto de patrones de conectividad-8, por otro lado, un patrón de conectividad-8 no puede estar con de conectividad-4. La figura 3.12, es un ejemplo de patrón de conectividad-8. [2]

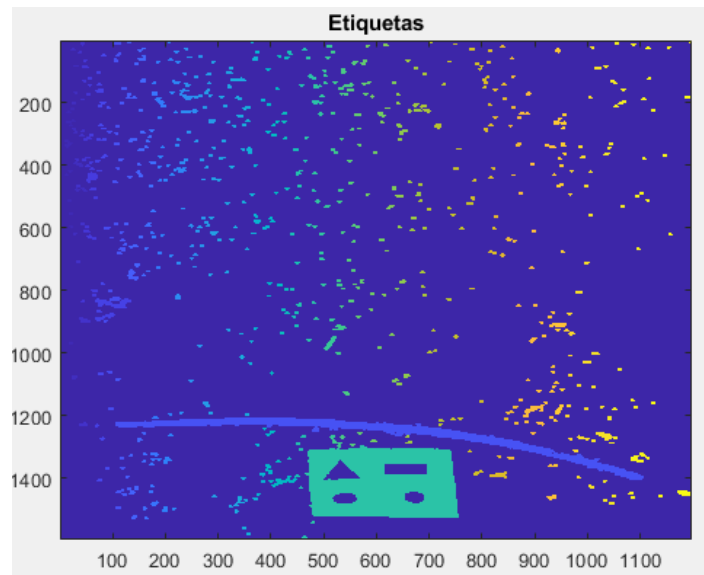
Figura 3.12 Patrón conectividad-8



Etiquetado de regiones

Para llevar a cabo dicho proceso, la imagen debe estar binarizada, y el objetivo es obtener una región. Actualmente existen diferentes algoritmos que llevan a cabo el proceso de etiquetado bajo el concepto de conectividad (conectividad-4 y conectividad-8) como puede ser el algoritmo descrito por Rosenfeld y Pfaltz (1966) [3]. Como se ha indicado previamente, estos algoritmos asignan a cada píxel una etiqueta, la cual, debe ser única para cada píxel de una región con el objetivo de dividir la imagen en regiones. Un ejemplo del etiquetado de este proyecto puede verse en la figura 3.13

Figura 3.13 Etiquetado de componentes



3.4 Identificación de regiones

La descripción de las regiones tiene como propósito llegar a su identificación o caracterización, a través de ciertos recursos y herramientas. Dependiendo del criterio o del punto de vista a utilizar, una región puede describirse de distintas maneras, puede definirse como un conjunto de puntos conectados entre sí, es decir, que a través es de un punto de la región pueda conectarse a otro sin abandonar la misma, o también por el número de huecos que presentan, a estas propiedades se le denominan propiedades topológicas. Sin embargo, si lo que obtenemos son valores como área, perímetro, etc. Nos referimos a propiedades métricas. A continuación, vamos a describir más detalladamente estas propiedades.

Propiedades métricas

Las métricas son generalizaciones de la distancia Euclídea, así una propiedad métrica cambiará si el plano de la figura se distorsiona. Algunas de las principales son las siguientes que se describen brevemente a continuación:

1. Área: El área A es el número de píxeles contenidos dentro de su frontera, por lo tanto, su valor dependerá de que resolución tenga la imagen.
2. Perímetro: El Perímetro P, es la longitud de su frontera, que se puede obtener a partir del código de cadena de la frontera. Aunque a veces se usa el perímetro como descriptor, su aplicación más usual se da en la obtención de la compatibilidad de una región que se define como P^2/A . Éste es un valor sin dimensiones que es mínimo para una región en forma de disco. Su expresión es la siguiente

$$P = \sum_i \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2} \quad (7)$$

3. Centro de gravedad (\bar{x}, \bar{y}) : Es un único punto representativo de la región y se obtiene como sigue,

$$\bar{x} = \frac{1}{A} \sum_i x_i, \quad \bar{y} = \frac{1}{A} \sum_i y_i \quad (8)$$

Como se puede observar a partir de la ecuación 8, se le ha dado el mismo peso a todos los píxeles que intervienen, aunque de vez en cuando debe tenerse en mente, que puede resultar provechoso hacer uso de la intensidad en cada punto, lo que provocaría el siguiente cambio respecto a la expresión anterior:

$$\bar{x} = \frac{\sum_i f(x_i, y_i) x_i}{\sum_i f(x_i, y_i)}, \quad \bar{y} = \frac{\sum_i f(x_i, y_i) y_i}{\sum_i f(x_i, y_i)} \quad (9)$$

4. Orientación: La orientación es una propiedad muy importante a la hora de determinar el ángulo de giro de nuestra plataforma. Los ejes mayor y menor de una región, que se definen en términos de su frontera, son útiles para obtener la orientación. El cociente de las longitudes de estos ejes, llamado excentricidad de la región, es también un descriptor global importante de la forma del objeto.

Finalmente, se puede ajustar una recta a la región, esta línea determina también la orientación de la región, mientras que el ángulo de esta línea se puede obtener a partir del conjunto de ecuaciones siguiente:

$$S_x = \sum x_i, \quad S_y = \sum y_i, \quad S_{xx} = \sum x_i^2, \quad S_{yy} = \sum y_i^2, \quad S_{xy} = \sum x_i y_i \quad (10)$$

$$M_{xx} = S_{xx} - \frac{S_x^2}{A}, \quad M_{yy} = S_{yy} - \frac{S_y^2}{A}, \quad M_{xy} = S_{xy} - \frac{S_x S_y}{A} \quad (11)$$

La orientación viene dada por:

$$\phi = \tan^{-1} \left\{ \frac{M_{xx} - M_{yy} + \sqrt{(M_{xx} - M_{yy})^2 + 4M_{xy}^2}}{2M_{xy}} \right\} \quad (12)$$

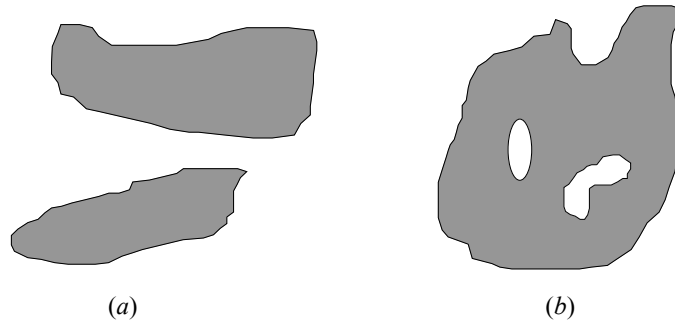
Propiedades topológicas

Estas propiedades se caracterizan por ser invariantes a ciertas deformaciones de las figuras en la imagen. Formalmente se define como una proyección continua uno a uno cuya inversa también es continua. Sin embargo, las propiedades topológicas no pueden involucrar nociones de distancia puesto que son distorsionadas y tampoco puede obtener nociones de distancia de forma indirecta como áreas, paralelismo de curvas, perpendicularidades líneas, etc.

Una de las descripciones topológicas de un conjunto más usadas normalmente es el número de sus componentes conexas. Una componente conexa de un conjunto es un subconjunto de dimensión máxima tal que cualesquiera dos de sus puntos pueden unirse por una curva continua (sin rupturas) perteneciente enteramente al subconjunto.

Una segunda propiedad topológica de interés es el número de huecos en la figura. Formalmente, el número de huecos en la figura es uno menos que el número de componentes conexas en el complemento de la figura. Siendo C el número de componentes conexas de una figura y H el número de huecos, definimos el número de Euler $E = C - H$, que también es una propiedad topológica. Como ejemplo, el número de Euler de las letras A y B son 0 y -1 respectivamente. En la figura 3.14 se muestra en (a) dos componentes conexas y en (b) una figura con dos huecos.

**Figura 3.14 (a) Figura con dos componentes conexas (b) Figura con dos huecos
(número de Euler-1)**



Las descripciones topológicas de las figuras encuentran aplicación en el análisis de la escena como un parámetro preliminar a otros descriptores más precisos. Por ejemplo, en reconocimiento de caracteres, si el número de Euler no coincide con un determinado patrón, el carácter ya no será comparado

Momentos geométricos

Los descriptores de Fourier que se usaban para describir una frontera eran insensibles a la traslación, rotación y cambio de escala. Cuando la región se da en términos de sus puntos interiores, podemos describirla mediante un conjunto de momentos que son invariantes a estos efectos.

Tomemos $f(x, y)$ como la intensidad del punto (x, y) en una región. El momento de orden $(p + q)$ para la región se define como:

$$m_{pq} = \sum_x \sum_y x^p y^q f(x, y) \quad (13)$$

Donde el sumatorio se toma sobre todas las coordenadas espaciales (x, y) de puntos de la región. El momento central de orden $(p + q)$ viene dado por:

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q f(x, y) \quad (14)$$

Los momentos *centrales normalizados* de orden $(p+q)$ se definen como:

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^\gamma} \quad \text{donde} \quad \gamma = \frac{p+q}{2} + 1 \quad \text{para } (p+q) = 2, 3, \dots \quad (15)$$

En este proyecto utilizamos los momentos invariantes propuestos por Hu, se puede obtener usando únicamente los momentos centrales normalizados de ordenes 2 y 3.:

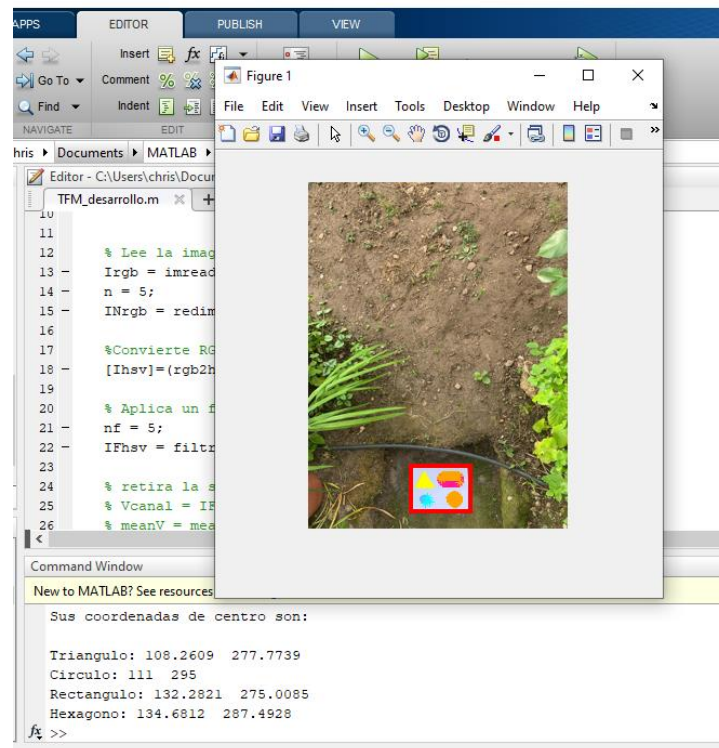
$$\begin{aligned} \phi_1 &= \eta_{20} + \eta_{02} \\ \phi_2 &= (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \\ \phi_3 &= (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \\ \phi_4 &= (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \\ \phi_5 &= (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + \\ &\quad (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \\ \phi_6 &= (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + \\ &\quad 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \\ \phi_7 &= (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + \\ &\quad (3\eta_{12} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \end{aligned} \quad (16)$$

Sin embargo, para que todos los momentos contribuyan por igual en la función de discriminación, y por tanto estén dentro del mismo orden de magnitud, a veces es necesario proceder a una normalización como la siguiente:

$$\phi'_n = \text{abs} \ln(\text{abs}(\phi_n)) \quad (17)$$

Para este proyecto, usamos la identificación de regiones para determinar la geometría de las figuras, así como, la posición en las que se encuentran. Estas verificaciones se encuentran detalladas técnicamente en el capítulo 5. En la imagen 5.15 podemos observar cómo indicamos la distancia a la que se encuentran los centroides de las figuras:

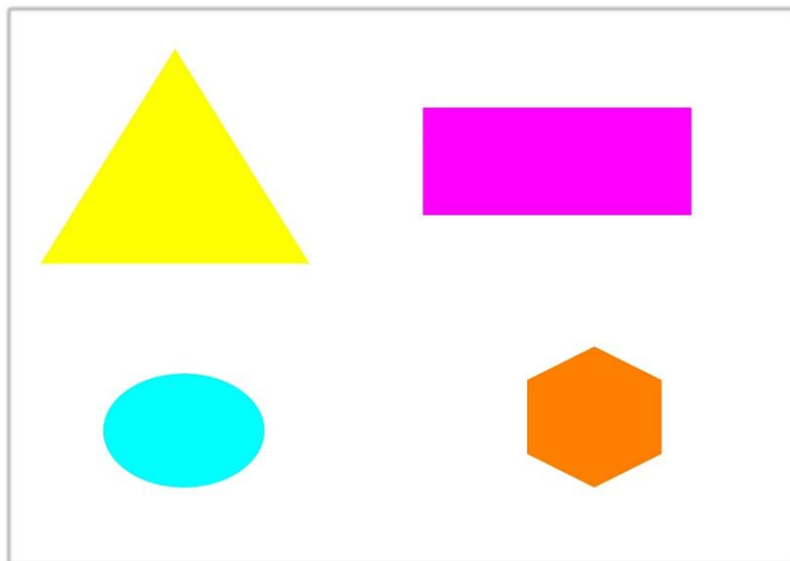
Figura 3.15 Identificación de figuras geométricas y distancias entre sus centroides



Chapter 4 - Modelo de identificación: Diseño del proyecto

En este trabajo, se ha llevado a cabo la determinación de figuras, mediante la extracción de colores y el reconocimiento de formas con la técnica de momentos de Hu. El primer paso para la extracción de colores consiste en binarizar las imágenes, para obtener el fondo negro y las figuras en blanco. Esta binarización se hace estableciendo un umbral para separar las regiones de la imagen en colores, blanca y negra. La imagen utilizada para la etapa de pruebas es un conjunto de figuras geométricas (ver figura 5.1).

Figura 5.1 Figuras para el entrenamiento



Esta figura es la que se emplea para entrenar el modelo. Se analiza la información de los píxeles de la imagen antes de establecer el umbral adecuado. El formato de la imagen se pasa antes al tipo `double`

```
Im = imread([path img]);  
%Muestra la imagen inicial con indicación de los píxeles  
figure(1); imshow(Im); title('Original'); impixelinfo
```

Inicialmente, se establece el primer umbral para obtener el primer componente del blanco. Se pasan los componentes R, G, y B de la imagen por el umbral inicial.

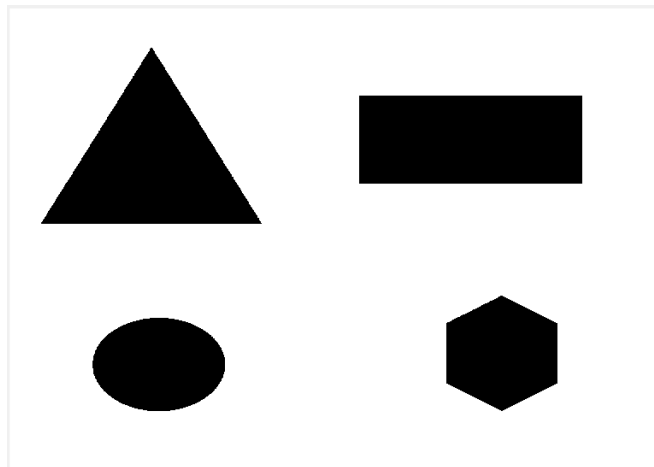
```
U1 = 200;  
Br1 = I(:,:,1) > U1; Bg1 = I(:,:,2) > U1; Bb1 = I(:,:,3) > U1;  
%Primer componente del □a blanca  
bw = and(and(Br1, Bg1), Bb1);
```

Asimismo, se recorta la imagen para establecer la plataforma en la región de interés.

```
[row, col] = find(bw);  
bounding_box = [min(row), min(col), max(row) - min(row) + 1,  
max(col) - min(col) + 1];  
%display with rectangle  
rect = bounding_box([2,1,4,3]);  
%rectangle wants x,y,w,h we have rows, columns, need to convert  
rectangle('Position', rect);  
cr = imcrop(Im, rect);
```

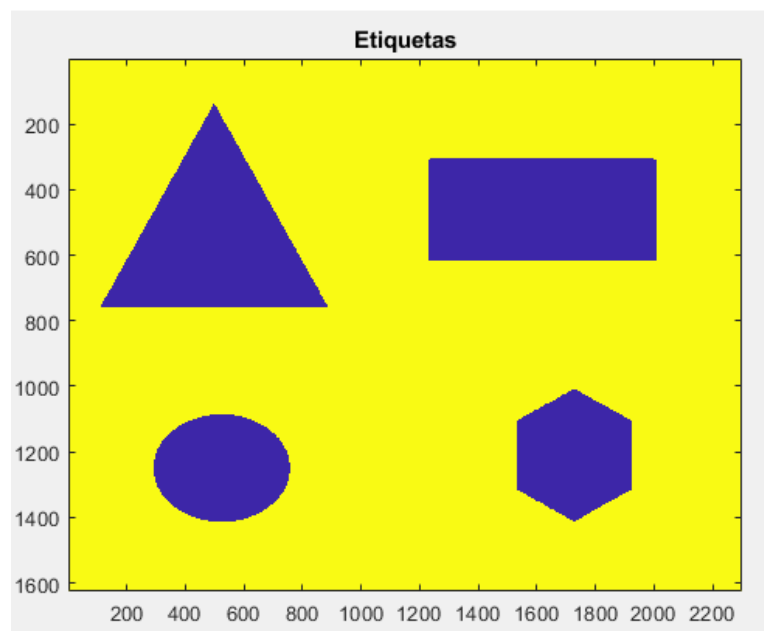
El umbral transforma la capa actual, es decir, la selección de la imagen en blanco y negro, donde los píxeles blancos representan los píxeles de la imagen cuyo valor se encuentra en el rango del umbral, y los píxeles negros representan los píxeles con el valor fuera del rango del umbral. Esta imagen binarizada se aprecia en la figura 5.2.

Figura 5.2 Figuras binarizadas



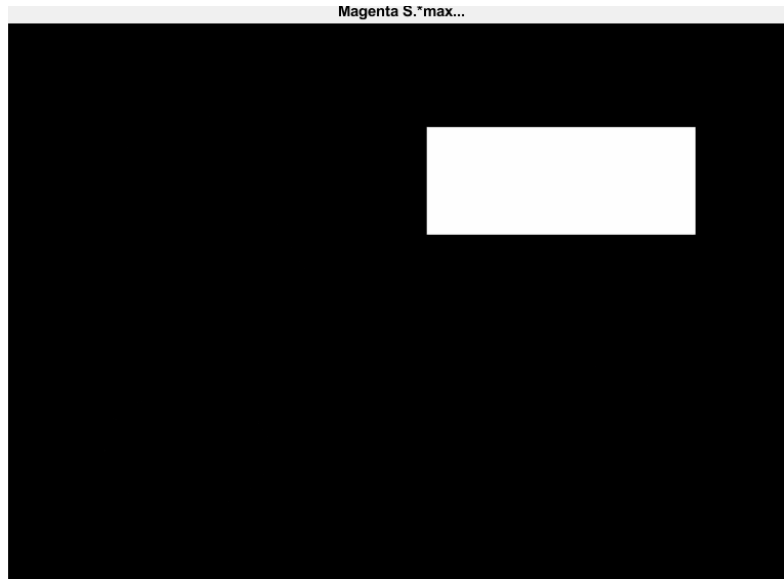
El siguiente paso, figura 5.3, consiste en etiquetar los valores de las imágenes. Esta etapa se denomina labeling en el procesamiento digital de imágenes. El etiquetado de componentes conectados escanea la imagen y se agrupan los píxeles en componentes en función de la conectividad de píxeles, es decir, todos los píxeles de los componentes conectados comparten valores de intensidad de píxel similares y están de alguna manera conectados entre sí.

Figura 5.3 Etapa de etiquetado



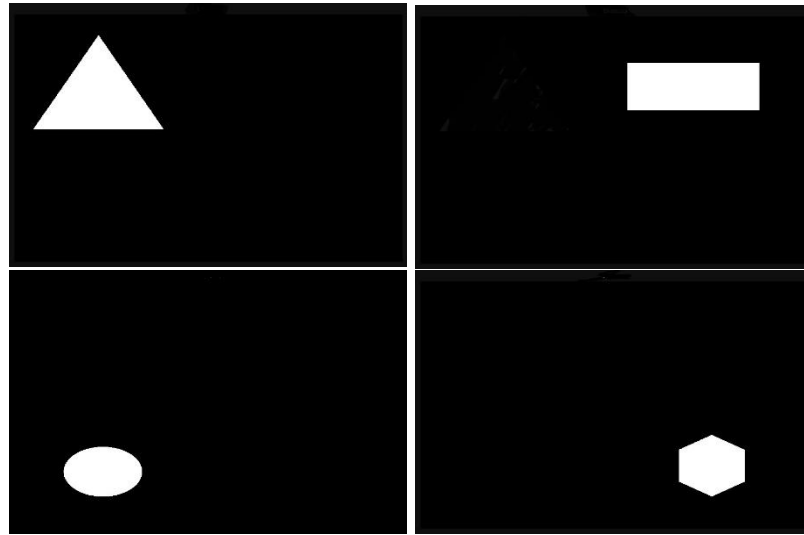
Luego se analizan las regiones encontradas en función de las propiedades de cada región. Esta tarea se lleva a cabo mediante la función de Matlab `regionprops`. Este procedimiento permite identificar las características de las figuras geométricas, como centroide, excentricidad, área, área convexa. Esta medida de formas para las figuras se hace en escala de grises. En esta etapa, figura 5.4, se obtienen las áreas blancas que permiten identificar los colores de cada figura.

Figura 5.4 Área blanca rectángulo



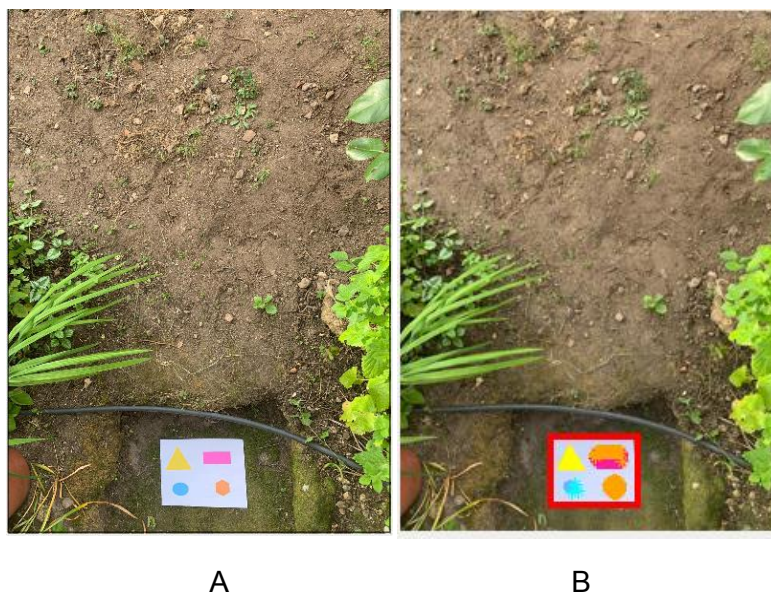
Estas operaciones, figura 5.5, se han realizado con figuras geométricas convencionales para evaluar el algoritmo de reconocimiento de figuras. En esta segunda etapa el algoritmo debe reconocer las figuras en espacios sometidos a iluminación natural y contrastes que dificultan el reconocimiento por extracción de color.

Figura 5.5 Áreas blancas figuras geométricas



El resultado permite identificar correctamente las figuras geométricas, conforme se aprecia en la figura 5.6-b. Las figuras geométricas son resaltadas con color cuando son identificadas, así como el marco de la plataforma. El entrenamiento del modelo es parte fundamental del algoritmo pues permite en las pruebas identificar las plataformas en diferentes entornos.

Figura 5.6 A) Imagen original B) Plataforma identificada



A

B

Asimismo, se identifican las distancias entre los centroides de las distintas figuras para verificar su correcta identificación. Esta información permite determinar las geometrías de las figuras en el siguiente paso. En esta etapa se realiza la identificación de las imágenes mediante sus descriptores geométricos:

```
function [s]= verifica(img)
    format long
    a = rgb2gray(img);

    bw = edge(a,'canny');
    bw = bwareaopen(bw,30);
    se = strel('disk',2);
    bw = imclose(bw,se);
    bw = imfill(bw,'holes');
    %L = bwlabel(bw);
    s = regionprops(bw, 'centroid');
    dt = regionprops(bw, 'area');
    cv = regionprops(bw, 'perimeter');
    dim = size(s);
    C=[];
    BW_filled = imfill(bw,'holes');
    boundaries = bwboundaries(BW_filled);
    figure(13); imshow(img, 'InitialMagnification', 'fit');
    hold on;
    % Se aplican los imgiterios geometricos
    for k=1:dim(1)
        bo = boundaries{k};
        dim = size(bo);
        for i=1:dim(1)
            diseno{k}(1,i) = sqrt((bo(i,2)-s(k).Centroid(1))^2 + (bo(i,1)-s(k).Centroid(2))^2 );
        end
        a = max(disenos{k});
        b = min(disenos{k});
        c = dt(k).Area;
        d = cv(k).Perimeter;

        % Ratio de areas de figuras
        cirmarca = a - b;
        cuamarca = c/(4*b^2);
        recmarca = c/(4*b*(a^2-b^2)^0.5);
        trimarca = (c*3^0.5)/((a+b)^2);
        elimarca = c/(a*b*pi);
        polmarca = c/(a*b*pi);
        %polmarca = (c*(a^2 - b^2 )^0.5)/(2*a^2*b);

        if cirmarca < 15
            % text(s(k).Centroid(1)- 20,s(k).Centroid(2), 'Circulo')
        elseif (cuamarca <1.05 ) && (cuamarca > 0.95)
            text(s(k).Centroid(1)- 20,s(k).Centroid(2), 'Cuadrado' )
        elseif (elimarca < 1.03) && (elimarca > 1.02)
            text(s(k).Centroid(1)- 20,s(k).Centroid(2), 'Elipse' )
        elseif (polmarca < 1.1) && (polmarca > 1)
            text(s(k).Centroid(1)- 20,s(k).Centroid(2), 'Poligono' )
        elseif (recmarca <1.05 ) && (recmarca >0.95)
            text(s(k).Centroid(1)- 20,s(k).Centroid(2), 'Rectangulo')
        elseif (trimarca <1.4 ) && (trimarca > 0.95)
            text(s(k).Centroid(1)- 20,s(k).Centroid(2), 'Triangulo')
        end
        C=[C, c]
    end
end
```

Chapter 5 - Resultados

Esta sección presenta algunas de las imágenes más importantes de la muestra de investigación, que se han convertido en la base para evaluar la tasa de éxito y el rendimiento del sistema de identificación y reconocimiento.

Estas imágenes han sido cuidadosamente seleccionadas para indicar las situaciones heterogéneas y diferentes que el sistema puede presentar. Por lo tanto, se muestran varias imágenes desde la perspectiva del proceso de adquisición para cubrir las condiciones más extremas posibles con respecto al proceso: imágenes tomadas a diferentes distancias o ángulos, con sombras, utilizando flash, etc.

Las imágenes han sido tomadas con la cámara del móvil iPhone, modelo XS MAX. Posee 2 cámaras que suman 24 megapíxeles entre ellas. Se determinó hacer las fotos con un móvil debido a que poseen cámaras pequeñas que son fácilmente implementables en drones debido a su pequeño tamaño y peso.

Por otro lado, la imagen IMG_60_1, ha sido tomada con otro móvil, el Oneplus 5T (16 megapíxeles), con el fin de comparar ambas resoluciones a la hora de identificar la plataforma.



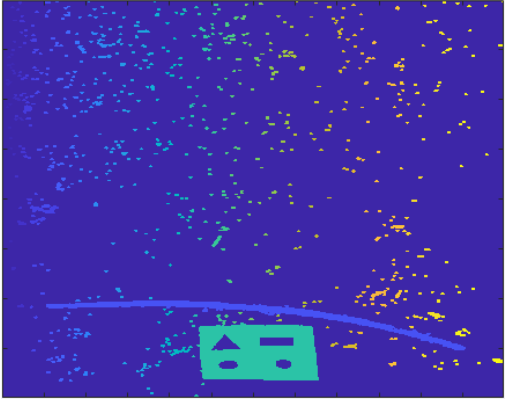

5.1 Ejemplo de imágenes


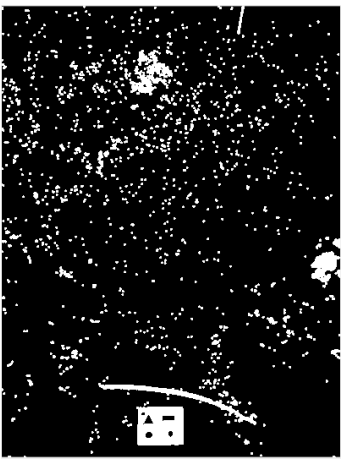
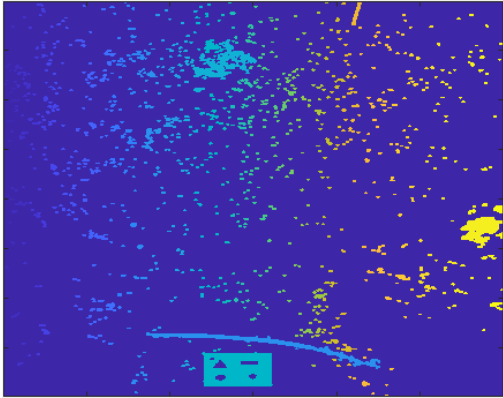

Para cada imagen seleccionada, se mostrará una serie de información, que se obtiene específicamente de las funciones de depuración implementadas por el algoritmo. Las imágenes que mostraremos son:


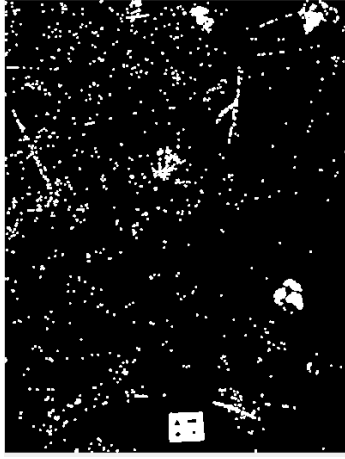
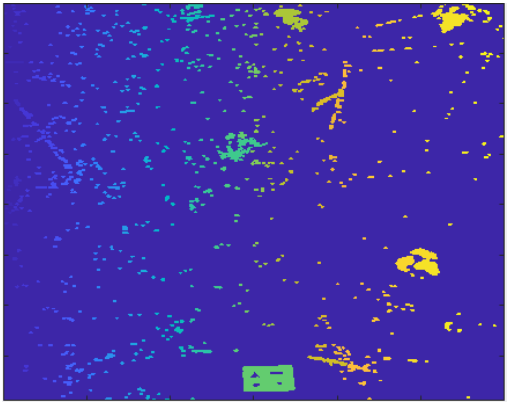

1. Imagen original de entrada
2. Etiquetas con regiones conexas
3. Imagen con todos los procesos de depuración aplicados
4. Imagen con plataforma identificada


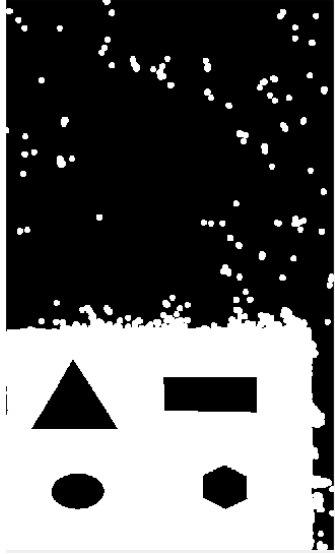
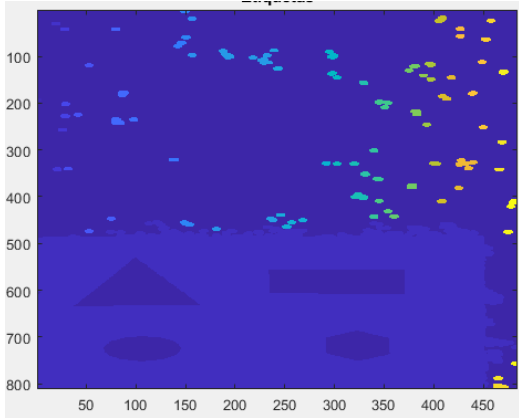

A la hora de identificar la plataforma si se ha identificado correctamente, colorea sobre las figuras detectadas. Ese coloreado superpone los colores que identifica por lo que según la luz puede que no lo superponga exactamente, pero si identifica bien la posición,



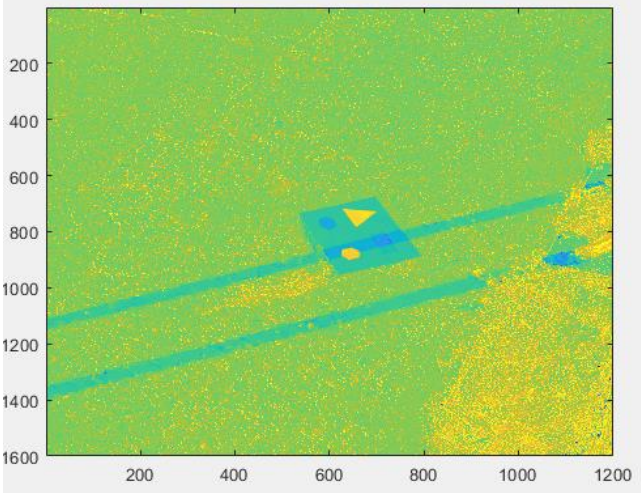

A continuación, expondré todas las imágenes con escalado de 1200 x 1600 pixeles para todas exceptuando las imágenes 40_1, 42_1 y 50_1 que tienen una dimensión de 900 x 1600:


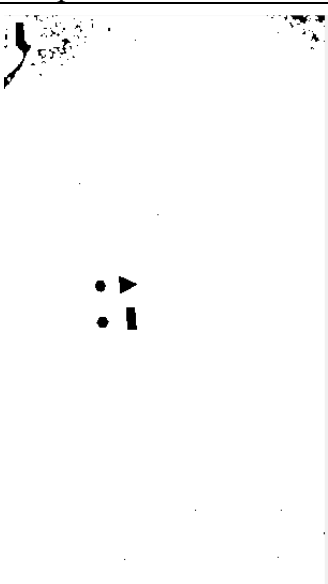
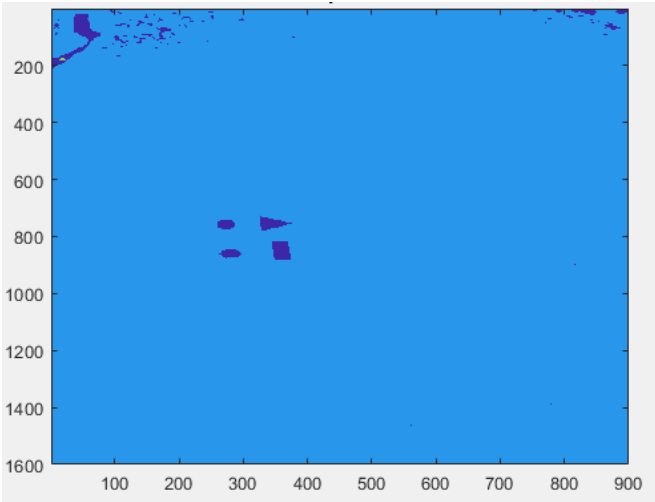

Nombre	Imagen original RGB/Regiones conexas	Imagen reconocida Depurada/Identificada
IMG_30_1.jpeg		
		


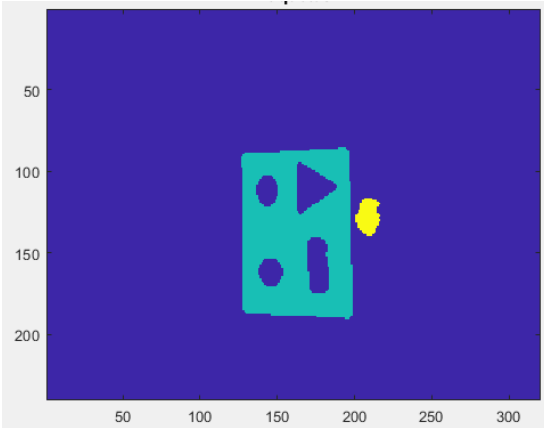


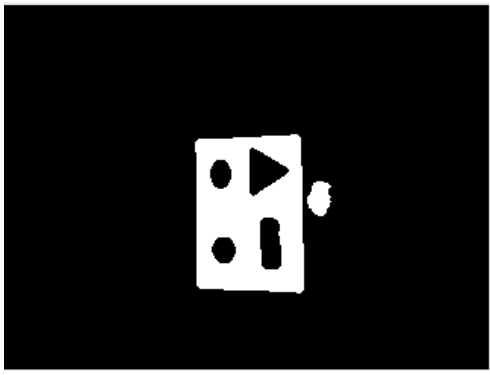
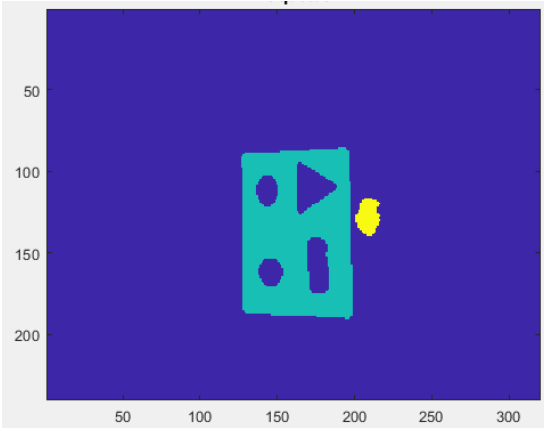

Nombre	Imagen original RGB/Regiones conexas	Imagen reconocida Depurada/Identificada
IMG_30_2.jpeg		
		

Nombre	Imagen original RGB/Regiones conexas	Imagen reconocida Depurada/Identificada
IMG_30_3.jpeg		
		


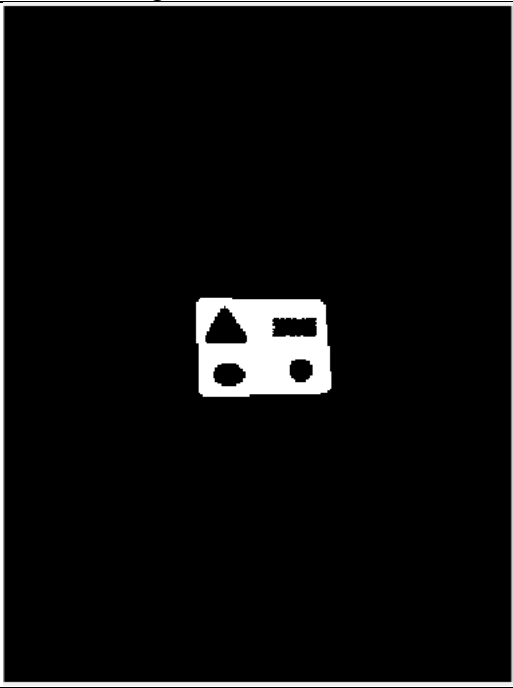
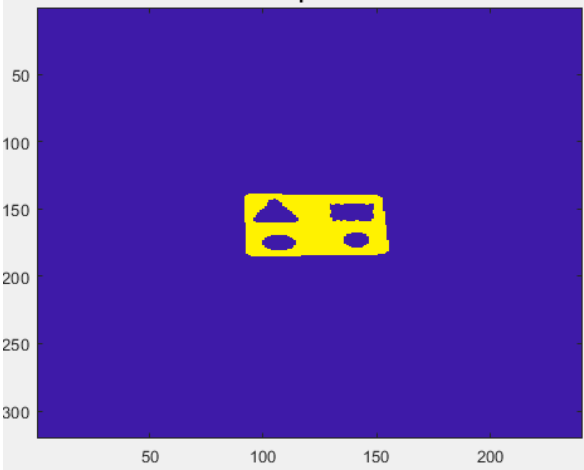
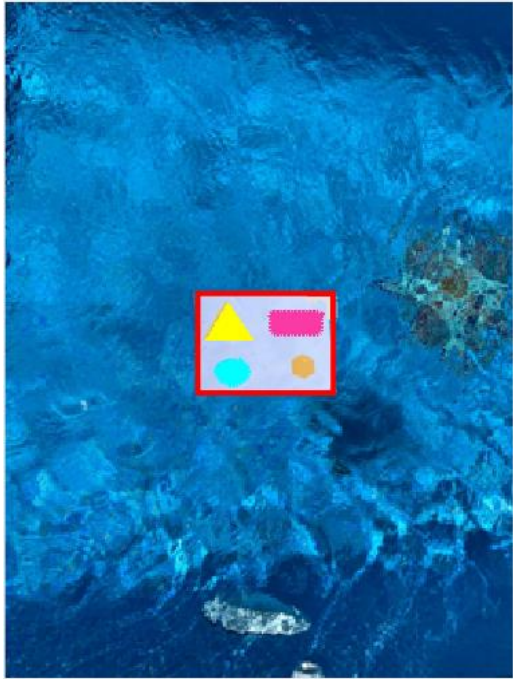
Nombre	Imagen original RGB/Regiones conexas	Imagen reconocida Depurada/Identificada
IMG_40_1.jpeg		
		


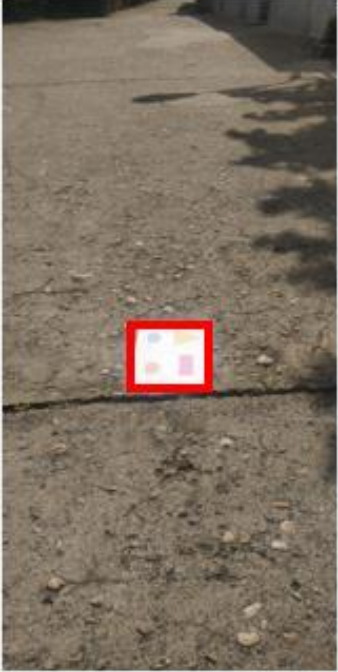
Nombre	Imagen original RGB/Regiones conexas	Imagen reconocida Depurada/Identificada
IMG_41_1.jpeg		
		

Nombre	Imagen original RGB/Regiones conexas	Imagen reconocida Depurada/Identificada
IMG_42_1.jpeg		
		

Nombre	Imagen original RGB/Regiones conexas	Imagen reconocida Depurada/Identificada
  		
		

Nombre	Imagen original RGB/Regiones conexas	Imagen reconocida Depurada/Identificada
IMG_50_1.jpeg		
		

Nombre	Imagen original RGB/Regiones conexas	Imagen reconocida Depurada/Identificada
IMG_50_2.jpeg		
		

Nombre	Imagen original RGB	Imagen reconocida Parcialmente identificada
IMG_60_1.jpeg		

5.2 Análisis de imágenes

Las imágenes expuestas en el apartado anterior tienen como objetivo revelar un determinado comportamiento del algoritmo. En relación con este aspecto se puede concluir lo siguiente:

1. **Imágenes IMG_30_1.jpeg e IMG_30_2.jpeg:** Estas imágenes han sido tomadas en un entorno de exteriores y a una cierta altura en relación con el lugar donde se encuentra la plataforma de aterrizaje, tal y como denota el tamaño de la figura respecto al tamaño total de la imagen. A través de estas dos imágenes, se trata de enfatizar la fiabilidad del algoritmo con relación a condiciones de altura estándar, puesto que el sistema es capaz de reconocer perfectamente la figura en dos

imágenes que reproducen con exactitud las condiciones que pueden darse en una imagen que se haya obtenido durante el transcurso de un vuelo rutinario.

2. **Imágenes IMG_30_3.jpeg e IMG_40_1.jpeg:** Representan condiciones totalmente opuestas en lo que a la altura se refiere. Ambas imágenes han sido obtenidas en un entorno de exterior con condiciones lumínicas similares y han sido reconocidas por el sistema. El objetivo es poner de manifiesto la versatilidad con la que trabaja el algoritmo desde el punto de vista de la distancia y la altura.

3. **Imágenes IMG_41_1.jpeg e IMG_42_2.jpeg:** Su objetivo es expresar nuevamente la versatilidad del algoritmo en condiciones de trabajo heterogéneas. En este sentido, debe enfatizarse que el algoritmo puede funcionar con ángulos de inclinación y con la plataforma en otra posición que no sea la centrada.

4. **Imágenes IMG_41_1.jpeg e IMG_43_1.jpeg:** Estas dos imágenes tratan de representar la capacidad que tiene el sistema para trabajar ante diversos tipos de condiciones lumínicas con el fin de garantizar el mayor número de aciertos ante situaciones de luz que se puedan dar a lo largo del día. La imagen IMG_41_1 se obtuvo en un entorno de exteriores en la que aparece una sombra sobre la plataforma. La imagen IMG_42_1 ha sido tomada en un entorno interior con el uso del flash, para producir reflejos sobre la plataforma. Por lo que observamos, ambas condiciones han sido detectadas y reconocidas por el algoritmo.

5. **Imágenes IMG_51_1.jpeg e IMG_51_2.jpeg:** Con estas imágenes pretendemos probar que la plataforma es identificada cerca y dentro de medios acuáticos. En la imagen 51_1 podemos observar que la plataforma se encuentra cerca del agua y la 52_2 se encuentra metida en una carpeta plástica y directamente situada sobre el agua. En ambas fotografías, determina bien la plataforma con respecto al medio que le rodea permitiendo identificarla de forma correcta. Con estos dos ejemplos podemos determinar que es posible situar la plataforma sobre

un vehículo acuático donde posteriormente aterrice un UAV o dron y no solo en un medio terrestre.

6. **Imagen 60_1.jpeg:** Esta imagen, a diferencias de las anteriores tomadas por el Iphone, está tomada por el Oneplus 5T. Las imágenes tomadas por este dispositivo, como se ve en el ejemplo, son muy difíciles de reconocer. En esta imagen podemos observar que, si es capaz de determinar la parte blanca de la plataforma, pero es incapaz de reconocer colores ni figuras. Por lo que podemos determinar que necesita de una buena resolución para poder identificar la plataforma, ya que, por el contrario, nos encontramos con situaciones idénticas a las mostradas en este ejemplo.

5.2 Tiempo de computación

El tiempo de ejecución de nuestro software es una variable importante, ya que, los vehículos no tripulados normalmente tienen una dinámica bastante rápida. Por tanto, es necesario que el lazo de control trabaje a una frecuencia alta, normalmente superior al Hz, es decir, por lo menos necesitamos generar señales de control a una frecuencia de 10 Hz (cada 0.1 s). Por lo que cada 0,1s deberíamos: tomar imagen, procesarla, calcular la posición relativa a la imagen y generar las órdenes a los motores del dron. Cuanto más grande es el vehículo más lento es su dinámica y por tanto este requisito se relaja.

Para poder generar las señales de guiado que alimentan al lazo de control es necesario el procesado de imagen que hemos desarrollado en este proyecto. Por lo tanto, el tiempo de proceso de la imagen limita la frecuencia del lazo de control.

Por todos estos motivos, entendimos que era importante contabilizar ese tiempo de ejecución para posteriormente tenerlo en cuenta a la hora de implementarlo en el UAV y saber cuánto tiempo consume el análisis de las imágenes.

Para la estimación del tiempo de ejecución, se realizaron 5 ejecuciones por los dos tipos de imagen según su resolución (1200 y 900 x 1600), y se obtuvo la media aritmética a partir de estos resultados parciales. Las imágenes con resolución 900 x 1200 tienen un tiempo de ejecución de 2,543468 segundos y de 2,551398 para dimensiones de 1200 x 1600.

Con estos resultados podemos observar que no existen casi diferencias de tiempos entre las imágenes de distintas resoluciones y que esta estimación es un poco excesiva para que sea eficiente cuando se aplique en un entorno real.

Chapter 6 - Conclusión y líneas futuras

En este trabajo se ha desarrollado un algoritmo para el reconocimiento de imágenes. Las figuras empleadas son de geometrías conocidas, como el rectángulo, el triángulo, la elipse y el polígono. El objetivo inicial ha consistido en la detección de las figuras, basadas en la extracción de píxeles considerando sus colores fundamentales (RGB) y la técnica de momentos de Hu. Este algoritmo se ha puesto a prueba sobre imágenes en entornos cotidianos con altos niveles de contraste e iluminación.

En la etapa inicial se ha entrenado el algoritmo con imágenes de fondo homogéneo, en las que no aparecen ruidos, y los niveles de iluminación han sido controlados. La técnica de detección de figuras se hace por los colores característicos que presentan las figuras. Asimismo, el etiquetado se ha incluido en esta etapa para facilitar el cálculo de medidas geométricas de las figuras utilizadas. Los resultados de las pruebas efectuadas evidencian que las figuras se pueden detectar por el método de extracción de colores, cuando la iluminación no afecta la intensidad de los píxeles de las imágenes.

Asimismo, se implementa la técnica de momentos de Hu para corregir errores de detección del primer componente del algoritmo. Ambas técnicas han realizado la detección de figuras con alta precisión, por encima de 95% en la etapa de entrenamiento del algoritmo. En la etapa de pruebas, el algoritmo es puesto a prueba en entornos cotidianos en los que la iluminación es el punto crítico. La primera parte del algoritmo detecta las figuras en entorno de baja iluminación.

En la etapa inicial, el color detectado sobre escala de grises ha permitido segmentar las regiones e identificar contornos. El método de momentos para detección de formas es mucho más robusto en este tipo de escenarios. Sin embargo, en entornos agresivos presenta limitaciones, que son corregidas por el segundo componente del algoritmo. Las características geométricas de las figuras son la base del reconocimiento de las figuras en esta etapa final.

Uno de los principales problemas que nos encontramos es que se necesitan fotografías con alta resolución y buena iluminación que potencie los colores para ser identificada, ya que el resultado varía mucho dependiendo de la resolución, pasando a no ser reconocida si la imagen no tiene calidad.

Aunque el sistema funciona bien al adquirir imágenes a diferentes alturas, en un entorno real y eficiente, se recomienda realizar el proceso de adquisición a una distancia, altura e inclinación constantes. Es importante que el ángulo de inclinación de la persona y el porcentaje de su tamaño total sean constantes, ya que ayuda a asignar umbrales a ciertas resoluciones, de modo que el proceso de adquisición se pueda mantener dentro de un entorno controlado. Sin embargo, si por alguna razón las imágenes deben obtenerse a diferentes alturas y distancias, los resultados son muy buenos.

El entorno en el que se encuentra la plataforma tiene un impacto significativo: colores que se presenten en las figuras como el azul del agua o los reflejos con el color amarillo, sombreados, etc. Por lo tanto, es importante situar la plataforma en una superficie que sea lo más uniforme posible en términos de color y textura. Cabe destacar que la plataforma al tener un fondo blanco A4 rectángulo, ha sido provocado un mayor rendimiento del algoritmo, ya que, al binarizar y segmentar la imagen se forma una zona homogénea que es fácilmente reconocible para su posterior análisis de identificación.

A pesar de que los resultados son excelentes ante este tipo de situaciones dinámicas, el riesgo de que existan imágenes que no sean reconocidas aumenta en consideración, respecto al hipotético escenario que se daría si todos los parámetros estuvieran bajo valores controlados.

Para terminar, se expondrá una serie de líneas de mejoras que permitan una evolución y perfección del proyecto:

1. El principal trabajo de futuro es incorporar nuestro software a situaciones reales con fotos tomadas por drones para poder comprobar que se mantiene la misma efectividad, y aplicar mejoras.
2. Nuestro principal inconveniente es la imposibilidad de detención de la plataforma con imágenes de resolución baja. Por lo tanto, como posible mejora es elaborar un algoritmo que permita determinar que resolución debe tener las imágenes para que se tomen con la resolución correspondiente al entorno para una mayor efectividad de reconocimiento.
3. Añadir nuevos elementos geométricos y colores que aumenten la convicción dado que introducir nuevos elementos acortan el margen de equivocación debido a que es muy difícil que se dé el caso de que se den las mismas condiciones que en nuestra plataforma.
4. El tiempo de cómputo es alto, por lo que es importante que sea más bajo pues no permitirían trabajar con una frecuencia de control del orden del Hercio. Por lo tanto, como línea de futuro gran parte del código de los algoritmos se podría vectorizar mejorando la eficiencia y a nivel de hardware se podría disponer de un DSP especializado que acelere los cálculos.
5. Implementación de redes neuronales que permita mediante un entrenamiento una mayor efectividad a la hora de identificar la plataforma y a descartar errores.

Referencias

- Devine, P. G., & Sherman, S. J. (1992). Intuitive versus rational judgment and the role of stereotyping in the human condition: Kirk or Spock? *Psychological Inquiry*, 3(2), 153-159.
- Hodges, F. M. (2003). The promised planet: Alliances and struggles of the gerontocracy in American television science fiction of the 1960s. *The Aging Male*, 6(3), 175-182.
- Amazon Prime Air. Available online: <https://www.amazon.com/Amazon-Prime-Air/b?ie=UTF8&node=8037720011>. (accessed on 6 March 2018).
- Amazon Prime Air. Available online: <https://www.amazon.com/Amazon-Prime-Air/b?ie=UTF8&node=8037720011>. (accessed on 6 March 2018).
- Anitha G, Kumar RNG (2012) Vision based autonomous landing of an unmanned aerial vehicle. *Procedia Eng.* 2012, 38, pp. 2250–2256.
- AprilTag. Available online: <https://april.eecs.umich.edu/software/apriltag.html>. (accessed on 23 February 2018).
- Araar O, Aouf N, Vitanov I (2017) Vision based autonomous landing of multirotor uav on moving platform. *Journal of Intelligent & Robotic System* 2017, vol. 85 pp. 369–384.
- ArUco Markers Detection. Available online: http://docs.opencv.org/trunk/d5/dae/tutorial_aruco_detection.html. (accessed on 23 February 2018).
- Bohanec M (2009) Decision making: A computer-science and information-technology viewpoint. *Interdisciplinary Description of Complex Systems*, vol. 7(2), pp. 22-37.
- Bosch S, Lacroix S, Caballero F (2006) Autonomous detection of safe landing areas for an UAV from monocular images. *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2006.
- Campos Y, Sossa H, Pajares G (2016) Spatio-temporal analysis for obstacle detection in agricultural videos. *Applied Soft Computing*, vol 45, pp. 86-97.
- Cesetti A, Frontoni E, Mancini A, Zingaretti P, Longhi S (2010) A vision-based guidance system for uav navigation. *Journal of Intelligent and Robotic Systems*, 2010, vol. 57(1–4), pp. 233–257.

- Chaves SM, Wolcott RW, Eustice RM (2015) NEEC research: Toward GPS–denied landing of unmanned aerial vehicles on ships at sea. *Nav. Eng. J.* 2015, vol. 127, pp. 23–35.
- Chen J, Miao X, Jiang H, Chen J, Liu X (2017) Identification of Autonomous Landing Sign for Unmanned Aerial Vehicle Based on Faster Regions with Convolutional Neural Network. *IEEE International Conference on Chinese Automation Congress (CAC)*, pp. 2019-2114.
- Chouhan, S.S., Kaul, A., Singh, U.P. (2018). Soft computing approaches for image segmentation: a survey. *Multimed Tools Appl.* <https://doi.org/10.1007/s11042-018-6005-6>.
- Cocchioni F, Mancini A, Longhi S (2014) Autonomous navigation, landing and recharge of a quadrotor using artificial vision. *International conference on unmanned aircraft systems (ICUAS)*, pp. 418–429.
- Cruz JM, Sánchez B, Pajares G (2012) System for guiding an unmanned vehicle towards a platform using visual analysis. Patent n° 201001592, 2013.
- Davide F, Alessio Z, Alessandro S, Jeffrey D, Scaramuzza D (2017) Vision-based autonomous quadrotor landing on a moving platform. *Journal of Intelligent & Robotic Systems* February 2017, vol. 85, no. 2, pp. 369–384.
- Davies, E.R. (2018). *Computer Vision: principles, algorithms, applications, learning*. Academic Press-Elsevier, London.
- Desaraju V, Michael N, Humenberger M, Brockers R, Weiss S, Matthies L (2015) Vision-based landing site evaluation and trajectory generation toward rooftop landing. *Autonomous Robots* 2015, vol. 39, no. 3, pp. 445–463.
- Dillencourt MB, Samet H, Tamminen M (1992) A general approach to connected-component labeling for arbitrary image representations. *Journal of the Association for Computing Machinery* 39 (2), 252–280.
- Dougherty ER, Lotufo RA (2003) *Hands-on Morphological Image Processing*. SPIE Tutorial Texts in Optical Engineering Vol. TT5. SPIE Publications.
- Duda RO, Hart PE, Stork DG (2006) *Pattern classification* (2nd ed.) New York; New Delhi: Wiley.
- Forster C, Faessler M, Fontana F, Werlberger M, Scaramuzza D (2015) Continuous on-board monocular-vision-based elevation mapping applied to autonomous landing of micro

- aerial vehicles. Proc. in Robotics and Automation (ICRA), 2015 IEEE International Conference on. IEEE, pp. 111–118.
- García-Pulido JA, Pajares G, Dormido S, de la Cruz JM (2017) Recognition of a landing platform for unmanned aerial vehicles by using computer vision-based techniques. *Expert Systems with Applications*, vol. 76, pp. 152-165.
- Garrido-Jurado S, Muñoz-Salinas R, Madrid-Cuevas FJ, Marín-Jiménez MJ (2014) Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition* 2014, vol. 47, no. 6, pp. 2280–2292.
- Gil Fernández M, Ferre M, Ribero A (2016) Desarrollo de un sistema de aterrizaje autónomo sobre una plataforma de carga para un UAV tipo Parrot AR.Drone 2, pp. 1–5.
- Gui Y, Guo P, Zhang H, Lei Z, Zhou X, Du J, Yu Q (2013) Airborne vision-based navigation method for UAV accuracy landing using infrared lamps. *Journal of Intelligent & Robotic Systems*, vol. 72, no. 2, pp. 197, 2013.
- Guili X, Yong Z, Shengyu J, Yuehua CH, Yupeng T (2009) Research on computer vision-based for UAV autonomous landing on a ship. *Pattern Recognition Letters*, 30(6), pp. 600–605.
- Hermansson J (2010) Vision and GPS based autonomous landing of an unmanned aerial vehicle, pp. 19–25.
- Henriques JF, Caseiro R, Martins P, Batista J (2015) High-speed tracking with kernelized correlation filters. *IEEE Trans. Pattern Anal. Mach. Intell.* 2015, 37, pp. 583–596.
- Hu MK (1962) Visual Problem recognition by Moment Invariant. *IRE Trans. Inform. Theory*, IT-8, pp.179-187.
- Jiang T, Geller J, Ni D, Collura J (2016) Unmanned aircraft system traffic management: concept of operation and system architecture. *International Journal of Transportation Science and Technology*, vol. 5, no. 3, pp. 123–135.
- Johnson A, Montgomery J, Matthies L (2005) Vision guided landing of an autonomous helicopter in hazardous terrain. *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2005.
- Kalal Z, Mikolajczyk K, Matas J (2012) Tracking–learning–detection. *IEEE Trans. Pattern Anal. Mach. Intell.* 2012, 34, pp. 1409–1422.

- Kong W, Zhang D, Wang X, Xian Z, Zhang J (2013) Autonomous landing of an UAV with a ground-based actuated infrared stereo vision system. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, 3–7 November 2013; pp. 2963–2970.
- Kong W, Zhou D, Zhang Y, Zhang D, Wang X, Zhao B, Yan C, Shen L, Zhang J (2014) A ground-based optical system for autonomous landing of a fixed wing UAV. In Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS 2014) September 14-18, Chicago, IL, USA, pp. 1-8.
- Kong W, Hu T, Zhang D, Shen L, Zhang J (2017) Localization framework for real-time UAV autonomous landing: An On-Ground Deployed Visual Approach. *Sensors* 2017, 17, 1437.
- Koschan A, Abido M (2008) *Digital Color Image Processing*. John Wiley & Sons.
- Kyristsis S, Antonopoulos A, Chanielakis T, Stefanakis E, Linardos C, Tripolitsiotis A, Partsinevelos P (2016) Towards autonomous modular UAV missions: The detection, geolocation and landing paradigm. *Sensors* 2016, vol. 16, no. 11, 1844.
- Lange S, Sünderhauf N, Protzel P (2008) Autonomous landing for a multirotor uav using vision. Workshop Proc. of SIMPAR 2008 Intl. Conf. on Simulation, Modelling and Programming for Autonomous Robots, pp. 482-491.
- Lange S, Sunderhauf N, Protzel P (2009) A vision based onboard approach for landing and position control of an autonomous multirotor uav in gps-denied environments. *Advanced Robotics*, 2009. ICAR 2009. International Conference on. IEEE, 2009, pp. 1–6.
- Lee D, Ryan T, Kim HJ (2012) Autonomous landing of a vtol uav on a moving platform using image-based visual servoing. *Robotics and Automation (ICRA)*, 2012 IEEE International Conference on. IEEE, 2012, pp. 971–976.
- Lee H, Jung S, Shim DH (2016) Vision-based UAV landing on the moving vehicle. In Proceedings of the International Conference on Unmanned Aircraft System, Arlington, MA, USA, pp. 1–7.
- Li X (2013) A software scheme for UAV's safe landing area discovery. *AASRI Procedia* 2013, 4, pp. 230–235.

- Li Y, Wang Y, Luo H, Chen Y, Jiang Y (2012) Landmark recognition for UAV autonomous landing based on vision. *Application Research of Computers* (07), pp. 2780-2783.
- Lin S, Garratt MA, Lambert AJ (2017) Monocular vision-based real-time target recognition and tracking for autonomously landing an uav in a cluttered shipboard environment. *Autonomous Robots*, vol. 41, no. 4, pp. 881–901, 2017.
- Ling K (2014) Precision landing of a quadrotor uav on a moving target using low-cost sensors. Master's Thesis, University of Waterloo, Canada. UWSpace. <http://hdl.handle.net/10012/8803>. Available on-line: <https://uwspace.uwaterloo.ca/handle/10012/8803>. (accessed on 23 February 2018).
- Martínez C, Campoy P, Mondragón I, Olivares-Méndez MA (2009) Trinocular ground system to control UAVs. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, St. Louis, MO, USA, 10–15 October 2009, pp. 3361–3367.
- Maturana D, Scherer, S (2015) 3D convolutional neural networks for landing zone detection from LiDAR. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '15)*, pp. 3471–3478, IEEE, Washington, DC, USA, May 2015.
- Mercimek M, Gulez K, Mumcu TV (2005) Real object recognition using moment invariants. *IEEE Transactions on Pattern Analysis*, vol. 30, pp. 765-775.
- Nguyen PH, Kim KW, Lee YW, Park KR (2017) Remote marker-based tracking for UAV landing using visible-light camera sensor. *Sensors* 2017, 17, 1987.
- Nguyen PH, Arsalan M, Koo JH, Naqvi RA, Truong NQ, Park KR (2018) LightDenseYOLO: A Fast and Accurate Marker Tracker for Autonomous UAV Landing by Visible Light Camera Sensor on Drone. *Sensors* 2018, vol. 18, no. 6, 1703.
- Olson E (2011) AprilTag: A robust and flexible visual fiducial system. *IEEE International Conference on Robotics and Automation (ICRA)*, 2011.
- Otsu, N. (1979). A Threshold Selection Method from Gray-Level Histograms. *IEEE Trans. Sys., Man, and Cybernetics*, 9(1), 62-66.
- Pajares G, Cruz de la, JM (2007a) *Visión por Computador: imágenes digitales y aplicaciones*. RA-MA, Madrid.
- Pajares G, Cruz de la, JM (2007b) *Ejercicios resueltos de Visión por Computador*. RA-MA, Madrid.

- Pajares G (2015) Overview and current status of remote sensing applications based on unmanned aerial vehicles (UAVs). *Photogrammetric Engineering & Remote Sensing*, 81(4), pp. 281–329.
- Pajares, G.; García-Santillán, I.; Campos, Y.; Montalvo, M.; Guerrero, J.M.; Emmi, L.; Romeo, J.; Guijarro, M.; Gonzalez-de-Santos, P. *Machine-Vision Systems Selection for Agricultural Vehicles: A Guide*. *J. Imaging* 2016, 2, 34.

Chapter 7 - Introduction

It is well known how the use of images has multiple applications, in areas as diverse as medicine, agriculture, industry in general and industry 4.0 in particular, and in recent times in the field of connected systems where the advances are certainly significant.

Within the broad fields of application, it is worth focusing on a very specific one for the immediate future. As is well known, advances in the field of Unmanned Aerial Vehicles (UAVs) are set to play an important role in many areas of application and various topics of application (Haystacks 2015): agriculture, forestry, monitoring and exploration of disasters produced by fire, earthquakes, hurricanes, tsunamis, avalanches, nuclear leaks among others or for location, rescue, forest and animal inventories, surveillance for security purposes, road traffic control or photogrammetry among many other possible applications, including even the delivery of parcels by certain product distribution companies.

The UAVs are equipped with various sensor technologies for both navigation (GPS, Inertial Measurement Units, communication systems, altimeters, gyroscopes) and detection depending on the mission assigned. In this last case, we can highlight, for example, the LIDAR (Laser Imaging Detection and Ranging) systems, Radar and especially cameras operating in different ranges of the spectrum. In this sense, it is worth highlighting those of an infra-red nature and within these, the thermal ones, the multispectral ones, which capture radiation in a wide range of wavelengths and specifically the cameras operating in the visible range of the spectrum, commonly known as RGB cameras.

Figure 7.1 shows a flight sequence of a UAV, in this case a quadrapole, during different tests carried out under the SALACOM project (2013), which has given rise to this work. The UAV is equipped with a Galaxy S6 Smartphone to capture images with the camera of this device, whose resolution is 1440×2560 pixels and under the Android operating system. In the mentioned SALACOM project, two autonomous cooperative

and collaborative systems were developed. The UAV has the mission of exploring the operating environment to detect possible areas of polluting discharges on the water surface by capturing images, to provide this information to the autonomous vessel that has to go to the area of the discharge for containment and operation on it to avoid and in any case minimize the possible environmental impact. All this is done under the design and approach of a cooperative system between autonomous vehicles.

Figura 7.1 A flight sequence of a quad-rotor equipped with a mobile phone



As it is well known, the autonomy of any UAV is limited and in any case, even if it is still operational, at the end of the mission, both within the scope of the SALACOM project and in any type of operation, the UAV must be placed in a safe location. In the SALACOM project, landing must be done on a platform specifically designed and patented for this purpose (Cruz et al., 2012). This platform is shown in Figure 7.2. In (a) view from above and in (b) with the UAV resting on it. By means of the vision system on board the UAV itself, an identification procedure has been designed for its recognition based on image processing and the application of specific Computer Vision techniques, always under the consideration that these methods must operate in outdoor environments where the environmental conditions, mainly derived from uncontrolled lighting, are completely adverse, which makes its application extremely difficult in order to achieve satisfactory results.

Figura 7.2 Views of the patented landing platform



(A)



(B)

Within the scope of the SALACOM project, the landing must be made on the autonomous ship, which has the platform on board for the landing by means of its recognition through the vision system, figure 7.3. Obviously, in this case, the fact that the landing is made safely and precisely on it is a critical situation, since otherwise the UAV would fall into the water with its consequent deterioration and possible definitive loss. In any case, the landing must always be made safely to avoid landing in critical areas that could endanger the UAV itself and sometimes the mission.

Figura 7.3 UAV landing platform on board an autonomous ship.



However, as will be seen later in relation to the state of the art, this design entails intrinsic difficulties with regard to its recognition by the proposed computer vision techniques. Indeed, since the UAV operates in outdoor environments, lighting conditions are highly changeable, with brightness or sharp changes in tonality leading to misrecognition or even lack of recognition. In fact, Figure 7.2 itself shows how the black hue in (a) changes to blue in (b). In addition, due to such adverse changes in lighting and brightness, sometimes certain black or dark areas appear white, making recognition difficult. This is the main reason for the present work, which aims to design a platform with the highest possible robustness, from the point of view of its recognition through the application of specific image processing techniques for subsequent recognition in the field of Computer Vision.

In Computer Vision, a conceptual distinction must be made between what is known as High and Low Level Vision. Following the approaches expressed in Pajares and Cruz (2007a, b) or Davies (2018), low level vision refers to the inherent processes carried out in any application from image capture to scene interpretation, which already corresponds to the high level process.

More specifically, as far as the low level is concerned, one starts with the capture of images with digital cameras according to the application (Pajares et al., 2016). The sensor devices are digital, obtaining the matrices of quantified intensity values according to the sensor and the spectral band to which these devices are sensitive. In this project, the images are in the visible spectrum and therefore RGB. These images are pre-processed to improve their quality, among them operations to eliminate noise, correct problems caused by lighting and exposure times, among which we can mention image enhancement. All of them allow operations known as image segmentation to be carried out, the purpose of which is to identify and determine objects in the scene (González and Woods, 2008; Pajares and Cruz, 2007a; Davies, 2018).

This work addresses the problem of recognizing figures on a homogeneous basis (white background), which is ultimately the platform that is being recognized. Its design is

explained later. This problem is addressed in the stages of segmentation, identification of patterns associated with the shapes of objects and figure recognition once these patterns have been considered. In the first stage, pixel extraction techniques are applied based on their fundamental colors (RGB), which allows the identification of figures on a white background. The areas identified are labeled accordingly, in order to extract relevant properties from the figures contained on the base. These properties are determined according to the geometric characteristics that define them. The platform recognition process is established by a decision procedure based on the colour information and its geometric properties. The design of this platform is established based on the problems analyzed in the state of the art, in order to propose a platform design accompanied by a computer vision method, with high and low level techniques, from here on identified as an algorithm.

In the testing stage, the robustness of the proposal is verified in outdoor environments where lighting is a critical point. The first part of the algorithm detects the figures in low light environments. However, in aggressive environments it has limitations, which are corrected by the second component of the algorithm.

Chapter 8 - Conclusion and future lines

In this project, an algorithm for image recognition has been developed. The figures used are of known geometries, such as the rectangle, the triangle, the ellipse and the polygon. The initial objective has consisted in the detection of the figures, based on the extraction of pixels considering their fundamental colors (RGB) and the technique of moments of Hu. This algorithm has been tested on images in everyday environments with high levels of contrast and illumination.

In the initial stage the algorithm has been trained with homogeneous background images, in which no noise appears, and the illumination levels have been controlled. The technique of figure detection is done by the characteristic colors of the figures. Likewise, labelling has been included in this stage to facilitate the calculation of geometric measurements of the figures used. The results of the tests carried out show that the figures can be detected by the colour extraction method, when the illumination does not affect the intensity of the pixels in the images.

Also, the technique of Hu moments is implemented to correct detection errors of the first component of the algorithm. Both techniques have performed figure detection with high accuracy, above 95% in the training stage of the algorithm. In the testing stage, the algorithm is tested in everyday environments where lighting is the critical point. The first part of the algorithm detects figures in low light environments.

In the initial stage, the color detected on gray scale has allowed to segment the regions and identify contours. The method of moments for shape detection is much more robust in this type of scenario. However, in aggressive environments it has limitations, which are corrected by the second component of the algorithm. The geometric characteristics of the figures are the basis of the figure recognition in this final stage.

One of the main problems we find is that we need high resolution photographs with good lighting that enhances the colors to be identified, since the result varies greatly depending on the resolution, becoming unrecognizable if the image has no quality.

Although the system works well when acquiring images at different heights, in a real and efficient environment, it is recommended to perform the acquisition process at a constant distance, height and tilt. It is important that the person's tilt angle and percentage of total size are constant, as it helps to assign thresholds to certain resolutions, so that the acquisition process can be kept within a controlled environment. However, if for some reason the images must be obtained at different heights and distances, the results are very good.

The environment in which the platform is located has a significant impact: colours that are presented in the figures such as the blue of the water or reflections with the yellow colour, shading, etc. It is therefore important to place the platform on a surface that is as uniform as possible in terms of colour and texture. It should be noted that the platform, as it has a white A4 rectangle background, has resulted in a higher performance of the algorithm, since, by binarizing and segmenting the image, a homogeneous zone is formed that is easily recognizable for subsequent identification analysis.

Although the results are excellent in this type of dynamic situation, the risk of unrecognized images increases in comparison with the hypothetical scenario that would occur if all the parameters were under controlled values.

To conclude, a series of lines of improvements will be presented to allow the evolution and perfection of the project:

1. The main work for the future is to incorporate our software into real situations with photos taken by drones in order to check that it maintains the same effectiveness, and to apply improvements.

2. Our main disadvantage is the impossibility of stopping the platform with low resolution images. Therefore, as a possible improvement is to develop an algorithm to determine what resolution should have the images to be taken with the resolution corresponding to the environment for greater recognition effectiveness.
3. Adding new geometric elements and colors that increase the conviction since introducing new elements shortens the margin of error due to the fact that it is very difficult to have the same conditions as in our platform.
4. The calculation time is high, so it is important that it is lower because they would not allow to work with a control frequency of the order of the Hertz. Therefore, as a future line much of the code of the algorithms could be vectorized improving the efficiency and at hardware level could have a specialized DSP to accelerate the computations.
5. Implementation of neural networks that allows through training a greater effectiveness in identifying the platform and rule out errors